Sampling and Inference in Complex Networks

PhD thesis defense

of

Jithin K. Sreedharan

Inria, France (Team: MAESTRO)

Jury

Reviewers:

Nelly Litvak - University of Twente, The Netherlands

Don Towsley - University of Massachusetts, USA

Examinators:

Philippe Jacquet - Nokia Bell Labs, France

Alain Jean-Marie - Inria (MAESTRO), France

Advisor:

Konstantin Avrachenkov - Inria (MAESTRO), France

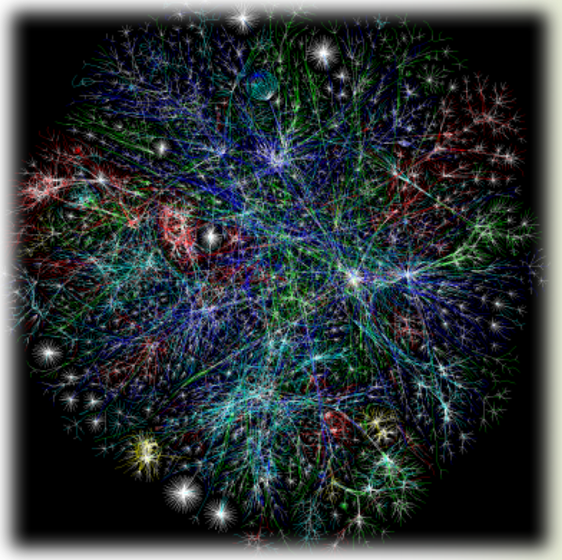
Date: December 2, 2016

Venue: Euler bleu, Inria Sophia Antipolis

Motivation

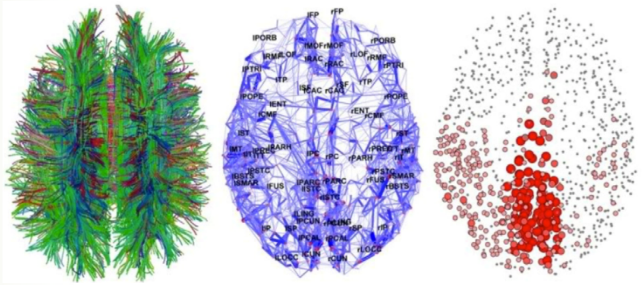
BIG data and Network Science

Technological networks,
e.g., Internet



Visual representation of the the Internet
from the Opte Project (www.opte.org)

Biological networks,
e.g., neural networks



Maria de la Iglesia-Vaya et al, “Brain Connections
– Resting State fMRI Functional Connectivity”, 2013

Social networks
e.g., online social network

Active users in Twitter:
30M (2010) -> 317M(2016) !

Motivation

Complex networks:

- Large size
- Sparse topology
- Small average distance (small-world)
- Many triangles
- Heavy tail degree distribution (scale-free phenomenon)

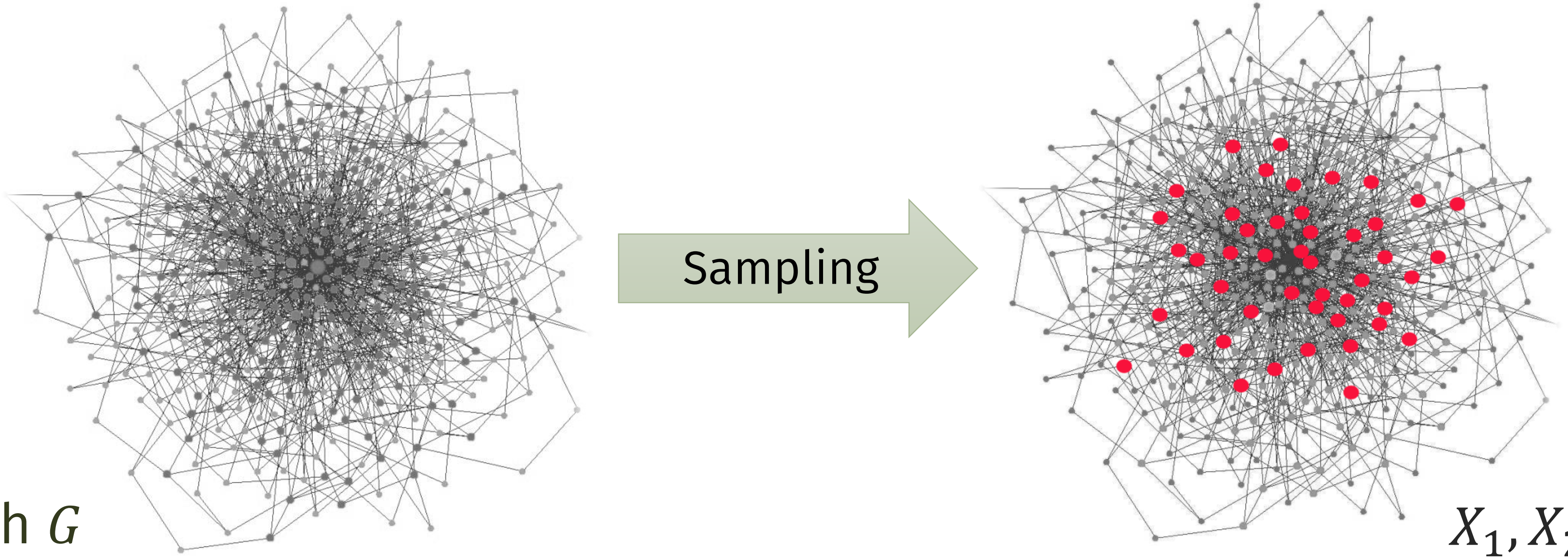
Some of the issues in the study of large networks

- What if the network is not known?

  Collecting data from the network takes time and huge resources (limited Application Programming Interface queries, e.g. Twitter)
- If the whole graph is collected, centralized processing has large memory requirements and long delays

Motivation

- Sampling: Collecting representative samples in a distributed way

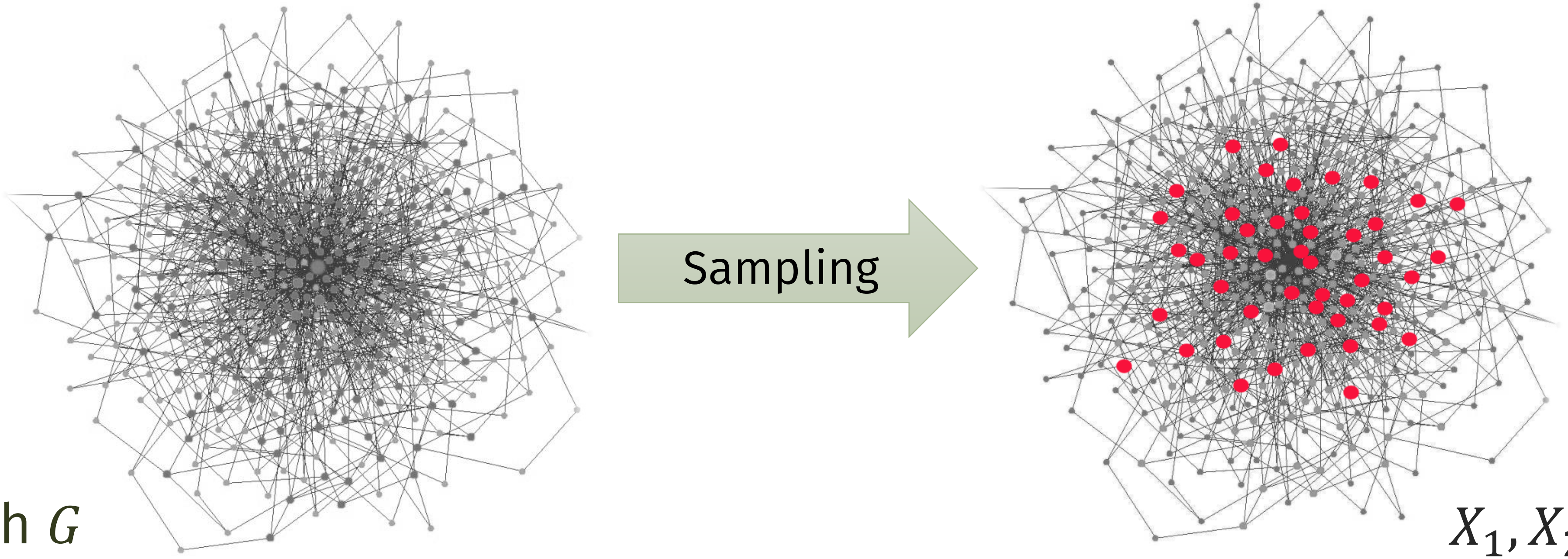


Graph G $\qquad$ $X_1, X_2, \ldots, X_k$

Samples: Independent?
Any stationary sequence e.g. node ID's, degrees, number of followers or income of the nodes in an online social network etc.

Motivation

- Sampling: Collecting representative samples in a distributed way



Graph G

$X_1, X_2, \dots, X_k$

- Estimate global property, $\mu(G)$ from $X_1, X_2, \dots, X_k$
- Make inference: How accurate the estimated value is, using posterior distribution of the estimator?

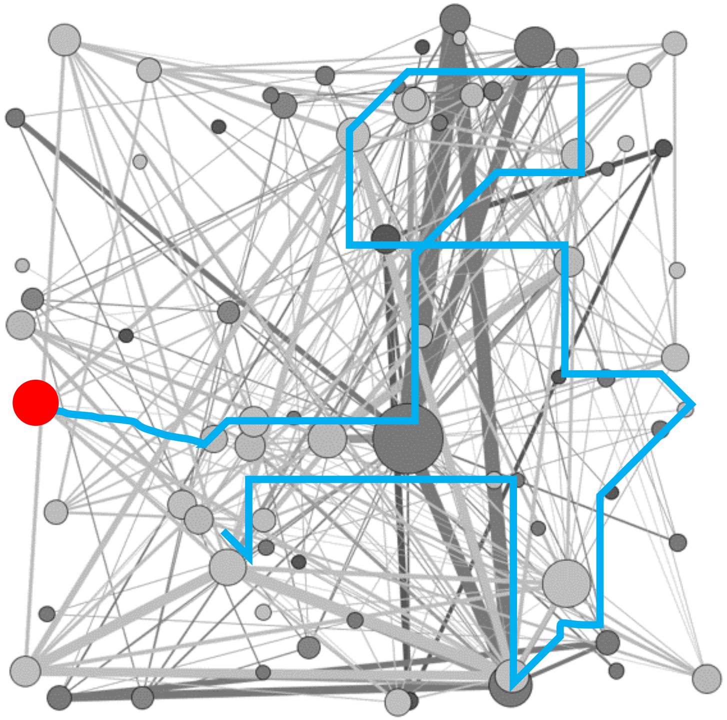
  Hypothesis testing on the graph using the samples collected

Topics Covered in this Thesis

1. Spectral Decomposition: Sampling in “Spectral Domain”

Topics Covered in this Thesis

1. Spectral Decomposition: Sampling in “Spectral Domain”
2. Network Sampling with Random Walk techniques

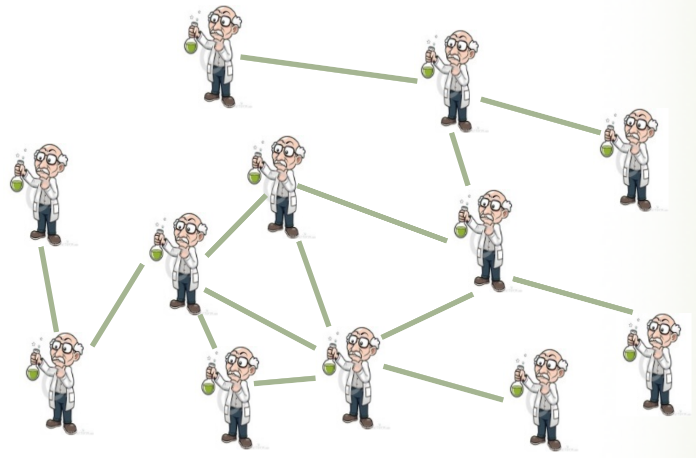


Les Misérables network

Topics Covered in this Thesis

1. Spectral Decomposition: Sampling in "Spectral Domain"
2. Network Sampling with Random Walk techniques
3. Extreme Value Theory and Network Sampling Processes

- All we have is the samples: $X_1, X_2, \ldots, X_n$
- Many networks are correlated, e.g., co-authorship n/w
- Extracting information from the correlated network
- Answers questions related to extremal events like
  - first time to hit a large node
  - clusters explored during the sampling process
  - .....



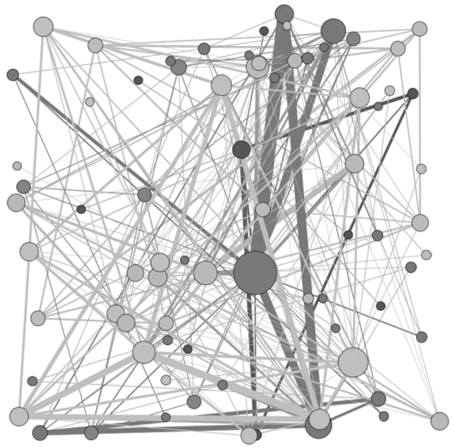
- Topic 1: Spectral Decomposition: Sampling in “Spectral Domain”
- Topic 2: Network Sampling with Random Walk techniques

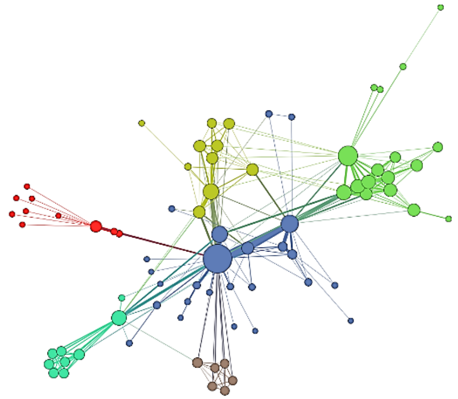
Motivation

Applications: Triangle counting, spectral clustering, asymptotic variance of random walks etc.

Graph Clustering:

- More difficult when graph is not known a priori
- An efficient solution is spectral clustering
- Requires knowledge of eigenvalues and eigenvectors of graph matrices





Question we address here

- Symmetric graph matrices like adjacency matrix $\mathbf{A}$, Laplacian matrix $\mathbf{L}$ of undirected graphs

$$\mathbf{A} = [a_{uv}], \quad a_{uv} = \begin{cases} 1, & \text{if } u \text{ is a neighbour of } v, \\ 0, & \text{otherwise.} \end{cases}$$

$$\mathbf{L} = \mathbf{D} - \mathbf{A}, \quad \mathbf{D} = \mathrm{diag}(d_1, \ldots, d_{|V|})$$

Degrees of nodes

Question we address here

- Symmetric graph matrices like adjacency matrix $\mathbf{A}$, Laplacian matrix $\mathbf{L}$ of an undirected graph $G = (V, E)$
- Eigenvalues: $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_{|V|}$

  Corresponding eigenvectors: $\mathbf{u}_1, \ldots, \mathbf{u}_{|V|}$

Problem

Scalable and distributed way to find dominant k eigenvalues $\lambda_1, \ldots, \lambda_k$ and the eigenvectors $\mathbf{u}_1, \ldots, \mathbf{u}_k$

Challenges in Classical Techniques for Finding the Spectrum

- Power iteration

$$\mathbf{b}_{\ell+1} = \frac{1}{\|\mathbf{b}_\ell\|}\mathbf{A}\mathbf{b}_\ell \quad \Rightarrow \quad \lambda_1 = \lim_{k\to\infty} \frac{\mathbf{b}_{k+1}^\intercal \mathbf{b}_k}{\|\mathbf{b}_k\|}, \qquad \mathbf{u}_1 = \lim_{k\to\infty} \frac{\mathbf{b}_k}{\|\mathbf{b}_k\|}$$

Drawback: Only principal components, orthonormalization

- Inverse iteration method

$$\mathbf{b}_{\ell+1} = \frac{1}{\|\mathbf{b}_\ell\|}(\mathbf{A} - \mu\mathbf{I})^{-1}\mathbf{b}_\ell \quad \Rightarrow$$

Closest eigenvalue to $\mu : \lim_{k\to\infty} \mu + \frac{\|\mathbf{b}_k\|}{\mathbf{b}_{k+1}^\intercal \mathbf{b}_k}$

Eigenvector : $\lim_{k\to\infty} \frac{\mathbf{b}_k}{\|\mathbf{b}_k\|}$

Drawback: Inverse calculation, orthonormalization

↓

With random walks in [Kempe & McSherry'08]

Complex Power Iterations: Central Idea

- Approach based on complex numbers

- Let $\mathbf{b}_t = e^{i\mathbf{A}t}\mathbf{b}_0$, solution of $\frac{\partial}{\partial t}\mathbf{b}_t = i\mathbf{A}\mathbf{b}_t$

  Adjacency matrix

  Initial vector

  Harmonics of $\mathbf{b}_t$ corresponds to eigenvalues

- Details: from spectral theorem,

Dirac-delta function

$$\frac{1}{2\pi}\int_{-\infty}^{+\infty} e^{i\mathbf{A}t}e^{-it\theta}dt \quad = \quad \sum_{j=1}^{n}\delta_{\lambda_j}(\theta)\mathbf{u}_j\mathbf{u}_j^\intercal$$

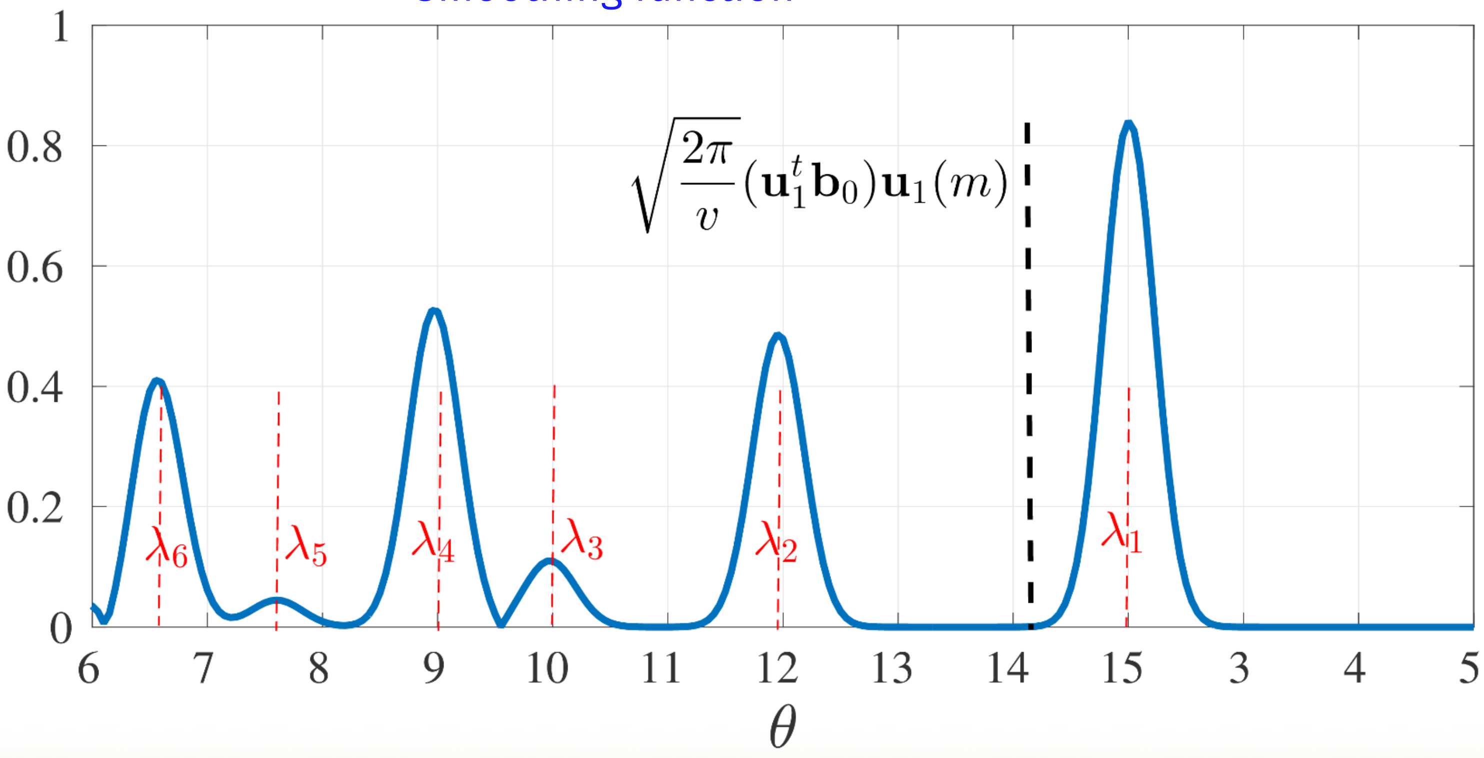
Complex Power Iterations: Smoothing and a sample plot

Idea of Gaussian smoothing:

$$\frac{1}{2\pi}\int_{-\infty}^{+\infty} e^{i\mathbf{A}t}\mathbf{b}_0\, \boxed{e^{-t^2 v/2}}\, e^{-it\theta}\,dt = \sum_{j=1}^{n}\frac{1}{\sqrt{2\pi v}}\exp\left(-\frac{(\lambda_j-\theta)^2}{2v}\right)\mathbf{u}_j(\mathbf{u}_j^{\intercal}\mathbf{b}_0)$$

Smoothing function

Smoothing parameter



Complex Power Iterations: Computing the Integral

Discretization:

Maximum no. of iterations

$$\mathbf{f}_{\theta} = \varepsilon \, \Re\left(\mathbf{b}_0 + 2 \sum_{\ell=1}^{\ell_{\max}} e^{-i\ell\varepsilon\theta} e^{-\ell^2\varepsilon^2 v/2} \mathbf{x}_\ell \right)$$

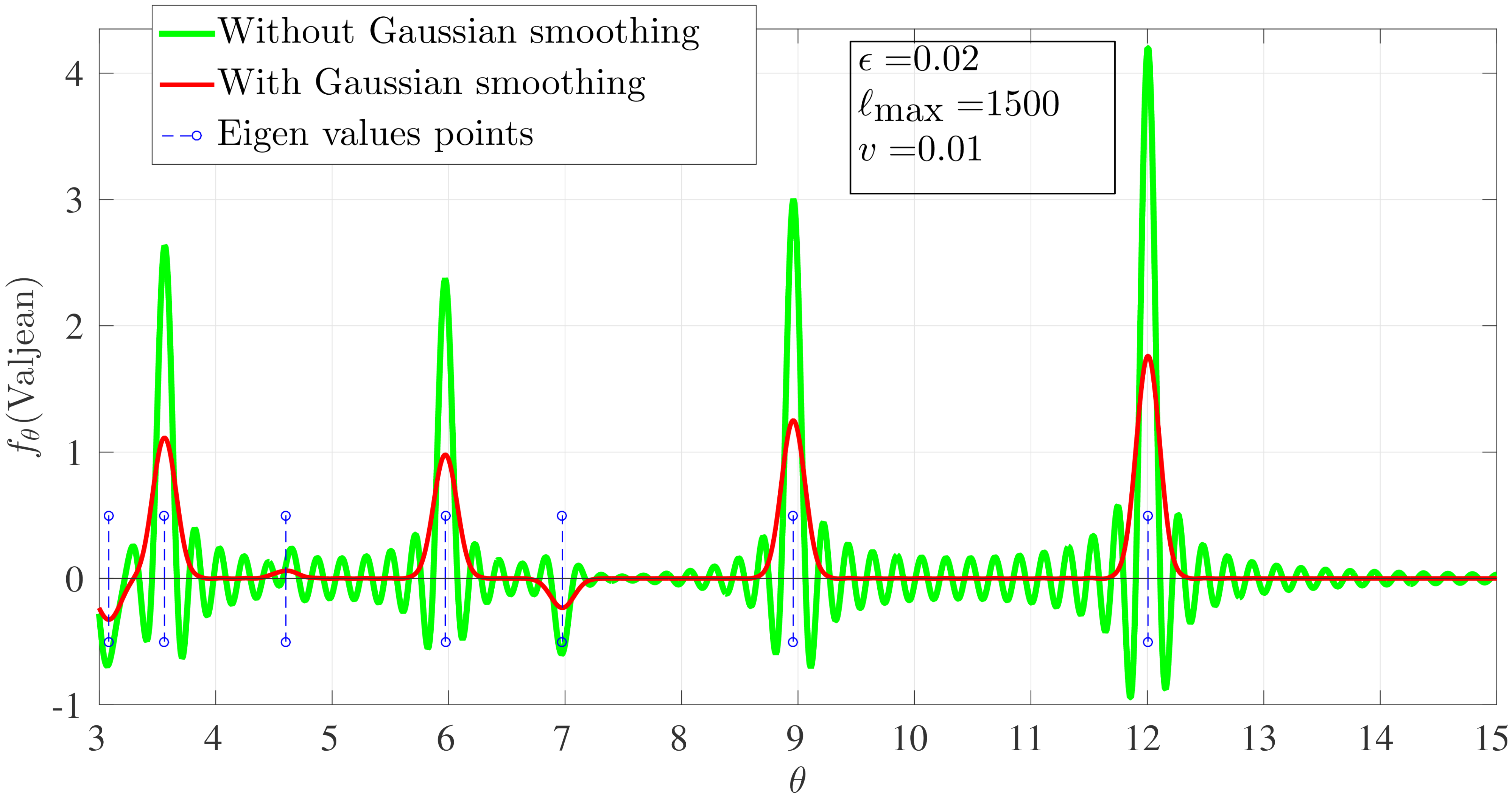
Interval length in Riemann sum

Approximation to $e^{i\varepsilon\ell\mathbf{A}}\mathbf{b}_0$

Approximations $e^{i\varepsilon\ell\mathbf{A}}\mathbf{b}_0$:

- First Order: $e^{i\mathbf{A}\ell\varepsilon} = (\mathbf{I} + i\varepsilon\mathbf{A})^{\ell}(1 + O(\varepsilon^2\ell))$
- Higher order: Numerical solution to $\frac{\partial}{\partial t}\mathbf{b}_t = i\mathbf{A}\mathbf{b}_t$ with $\mathbf{b}_0$ as the initial value. Use Runge-Kutta (RK) methods.

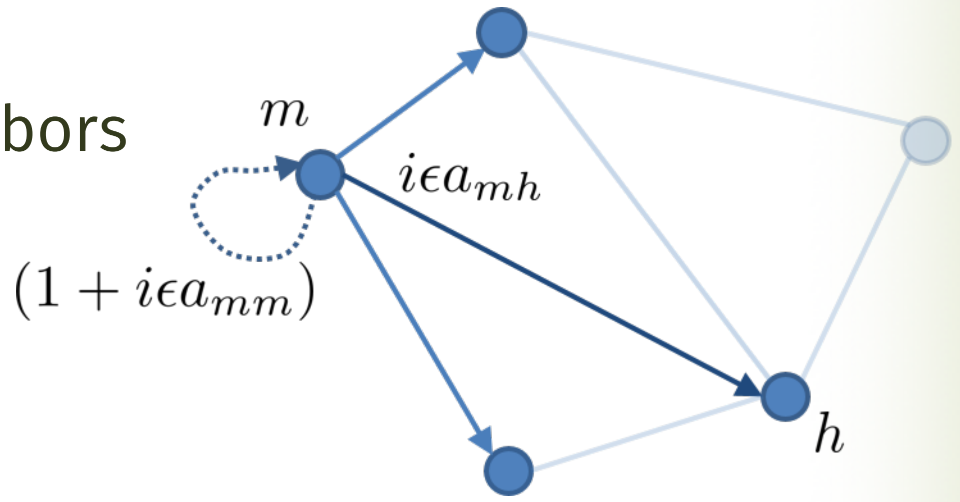
Gaussian Smoothing



Different Settings

1. **Centralized setting** : Adjacency matrix is fully known
2. **Our distributed approaches**
   - **Complex diffusion:** Asynchronous. Only local information available, communicates with all the neighbors

- Initialize node m with $\mathbf{b}_0(m)$
- Move weighted copy of fluid to all neighbors and to itself



Different Settings

1. **Centralized setting** : Adjacency matrix is fully known
2. **Our distributed approaches**
   - **Complex diffusion:** Asynchronous. Only local information available, communicates with all the neighbors

Inverse power iteration : For each λ

$$\begin{aligned} \text{delay} &= \operatorname{diam}(G) + 2\operatorname{diam}(G)\ell_{\max} \\ \text{no. of packets} &= |E||V|^2 + (|V||E| + |E|)\ell_{\max} \end{aligned}$$

Complex diffusion order-1: For all λ

$$\begin{aligned} \text{delay} &= \ell_{\max} + \operatorname{diam}(G) \\ \text{no. of packets} &= |E|\ell_{\max} + |V||E| \end{aligned}$$

Different Settings

1. **Centralized setting** : Adjacency matrix is fully known
2. **Our distributed approaches**
   - **Complex diffusion:** Asynchronous. Only local information available, communicates with all the neighbors
   - **Monte Carlo Gossiping:** Only local information, and communicates with only one neighbor.

   Basic idea:

$$
\begin{aligned}
\text{Let } \mathbf{x}_{k+1} &= (\mathbf{I} + i\varepsilon\mathbf{A})\mathbf{x}_k, \quad \mathbf{x}_0 = \mathbf{b}_0. \\
&= \mathbf{x}_k + i\varepsilon\mathbf{D}\,\mathbf{P}\,\mathbf{x}_k, \\
\mathbf{x}_{k+1}(m) &= \mathbf{x}_k(m) + i\varepsilon \underset{\text{Degree of node } m}{d_m} \mathbb{E}[\mathbf{x}_k(\xi_m)],
\end{aligned}
$$

$\mathbf{P} = \mathbf{D}^{-1}\mathbf{A}$

Rando walk t.p.m. matrix

ξ_m : Randomy selected neighbor of node m

Implementation with Quantum Random Walk (QRW)

We tried to solve a discretization of $\frac{\partial}{\partial t}\mathbf{b}_t = i\mathbf{A}\mathbf{b}_t$

Very similar to classic Schrödinger equation:

wave function

$$i\hbar\frac{\partial}{\partial t}\boldsymbol{\psi}_t = \mathbf{H}\boldsymbol{\psi}_t$$

Planck constant

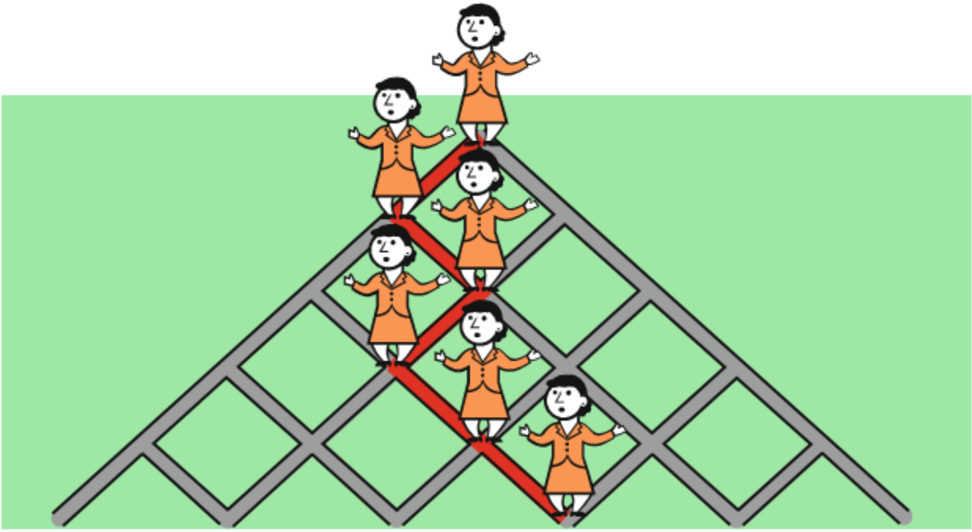
Hamiltonian operator

Continuous time QRW on a graph: $\boldsymbol{\psi}_t = e^{-i\mathbf{A}t}\boldsymbol{\psi}_0$

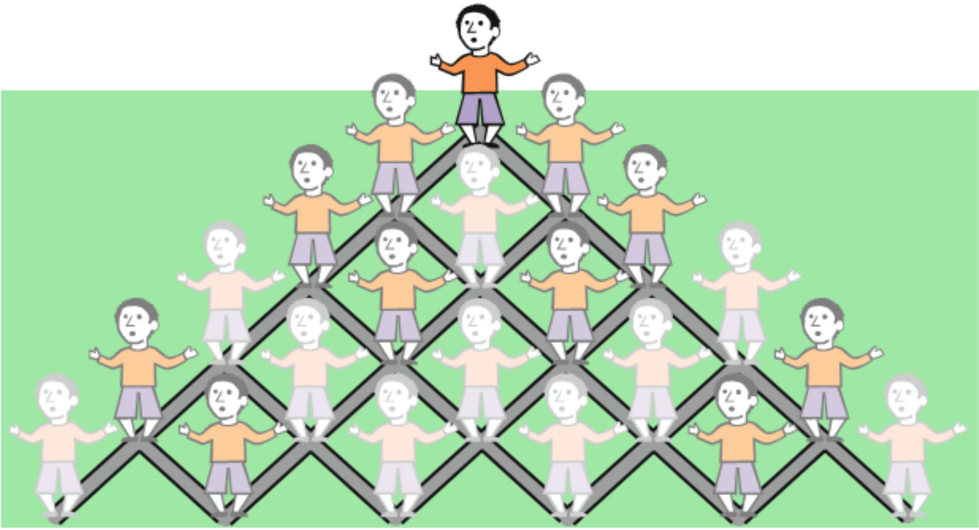
$\boldsymbol{\psi}_t$ is a complex amplitude vector $\{\boldsymbol{\psi}_t(i), 1 \leq i \leq n\}$.

When measured, the probability of finding QRW in node i at time t is $|\boldsymbol{\psi}_t(i)|^2$.

Sample Path Example



A sample path of classical RW



A sample quantum wave function of QRW

Rate of Convergence and Scalability

$$\int_{-\infty}^{+\infty} e^{i\mathbf{A}t}\mathbf{b}_0\, e^{-t^2 v/2} e^{-it\theta}\, dt$$

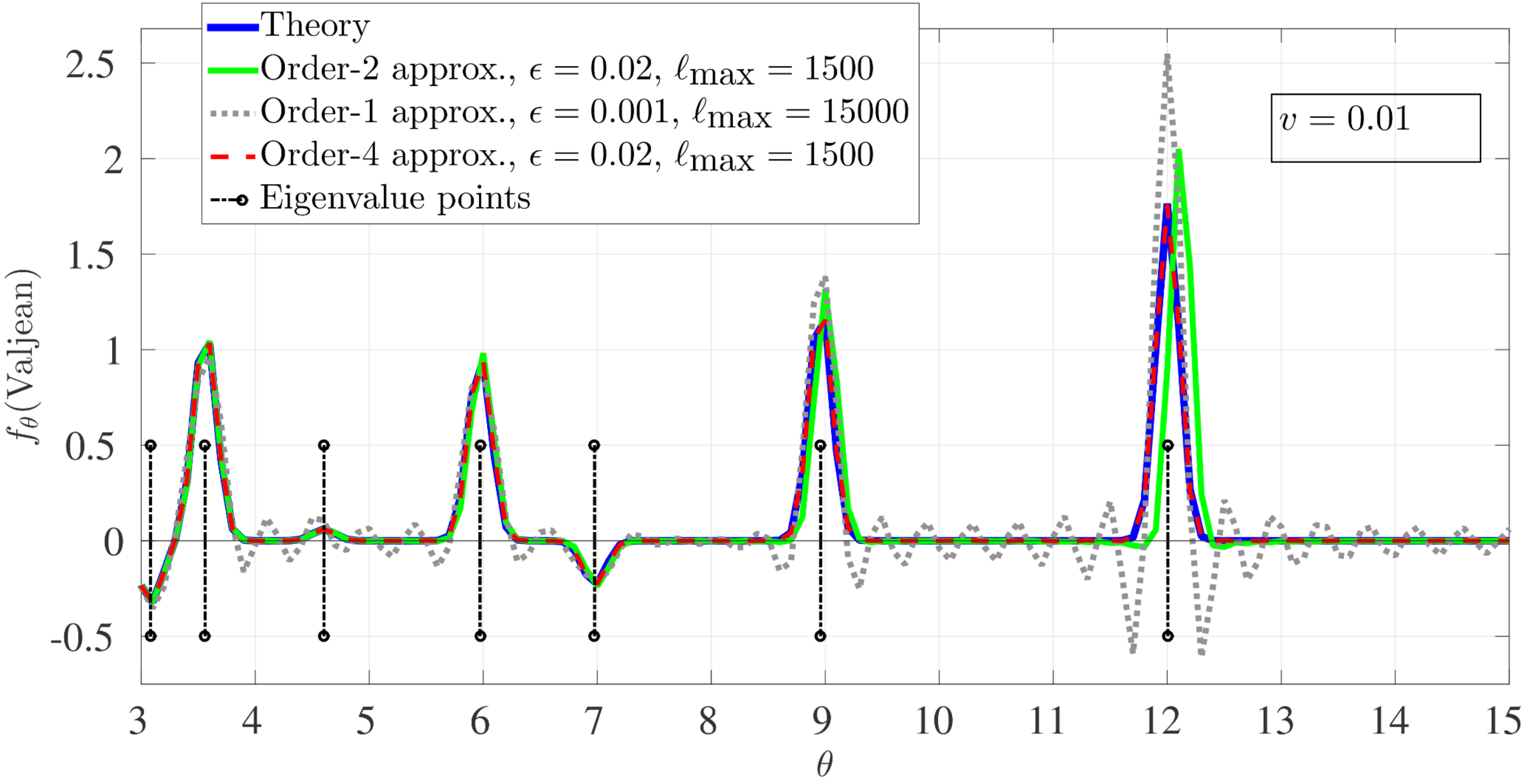
$$= \quad \varepsilon \Re \left(\mathbf{I} + 2 \sum_{\ell=1}^{\ell_{\max}} e^{i\ell\varepsilon\mathbf{A}} \mathbf{b}_0 e^{-i\ell\varepsilon\theta} e^{-\ell^2\varepsilon^2 v/2} \right) + \mathcal{O}\left(\lambda_1\, \varepsilon^2 \ell_{\max} \|\mathbf{b}_0\|\right)$$

No. of iterations $\ell_{\max}$ depends only on maximum degree.

Simulations on Real-World Networks

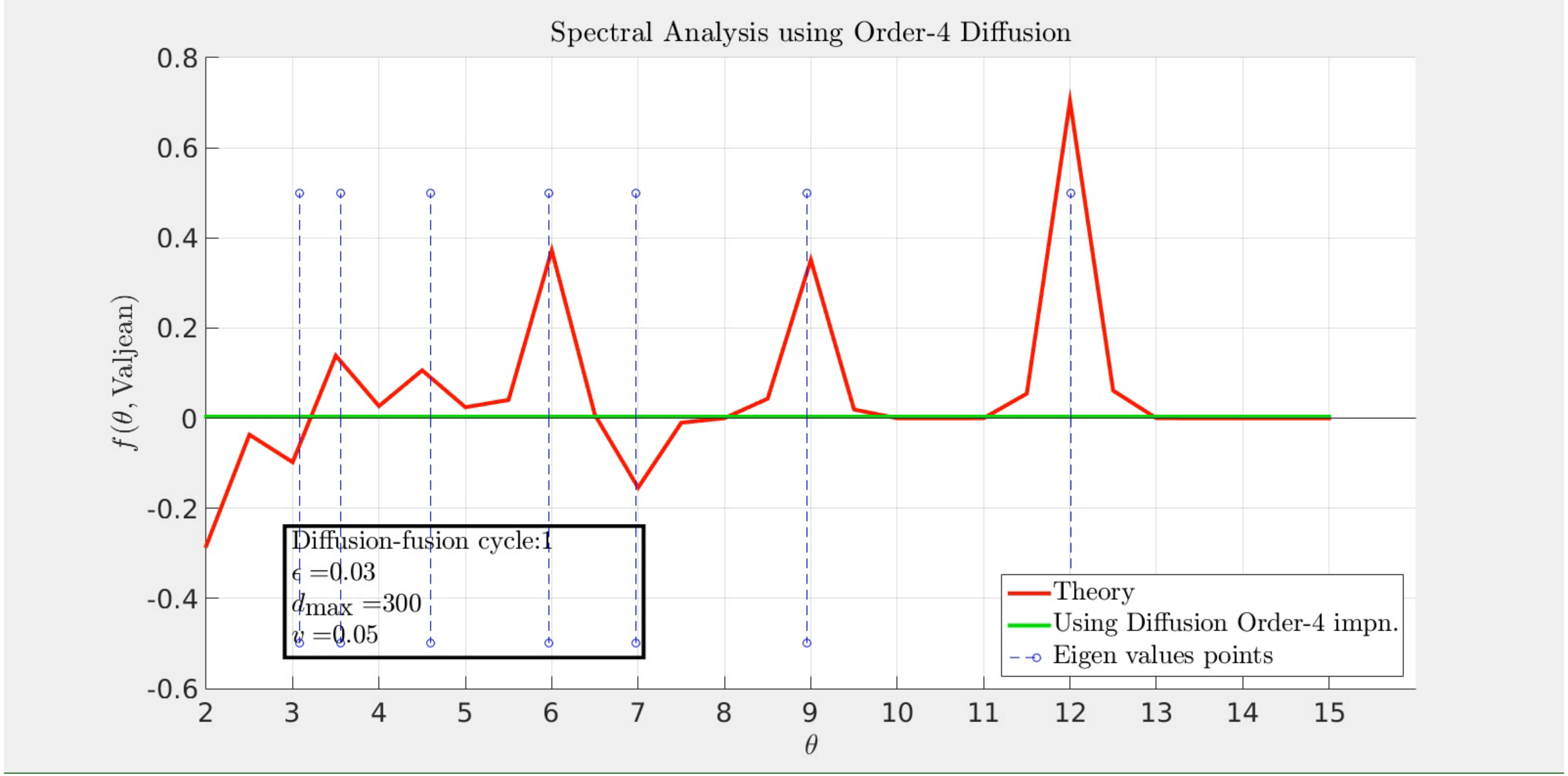
Spectral Decomposition: Sampling in “Spectral Domain”

Les Misérables network

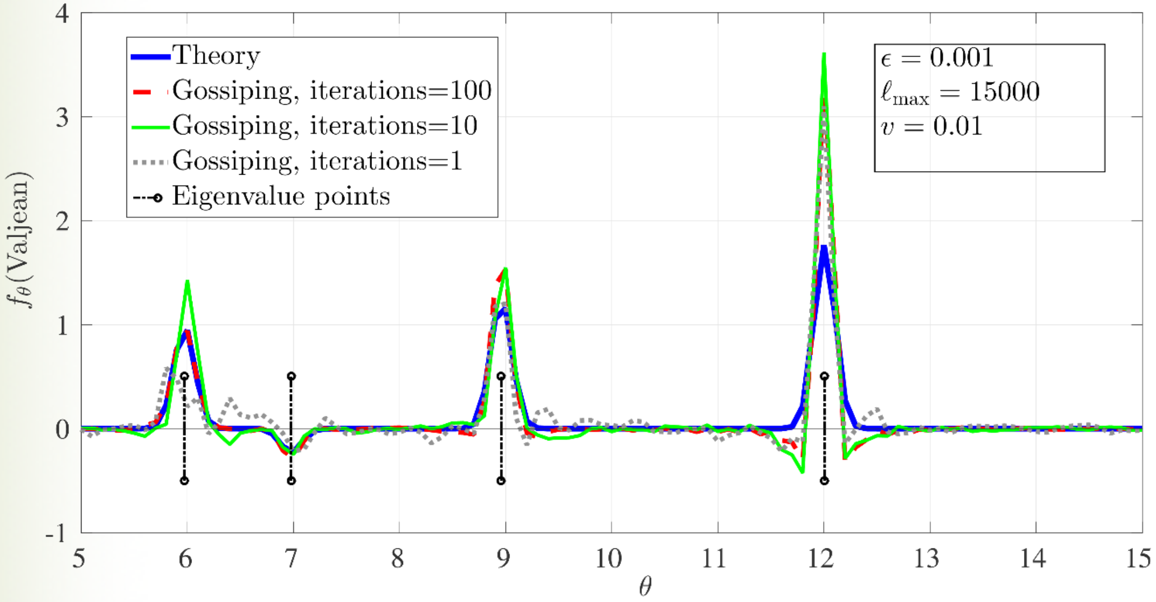


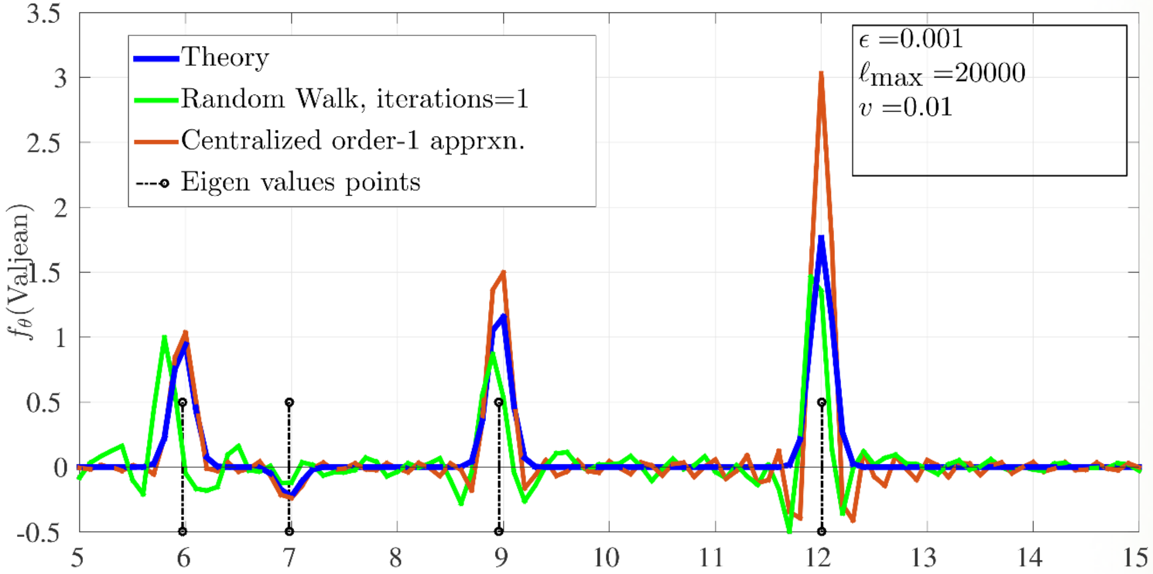
Complex diffusion

Les Misérables network



Les Misérables network



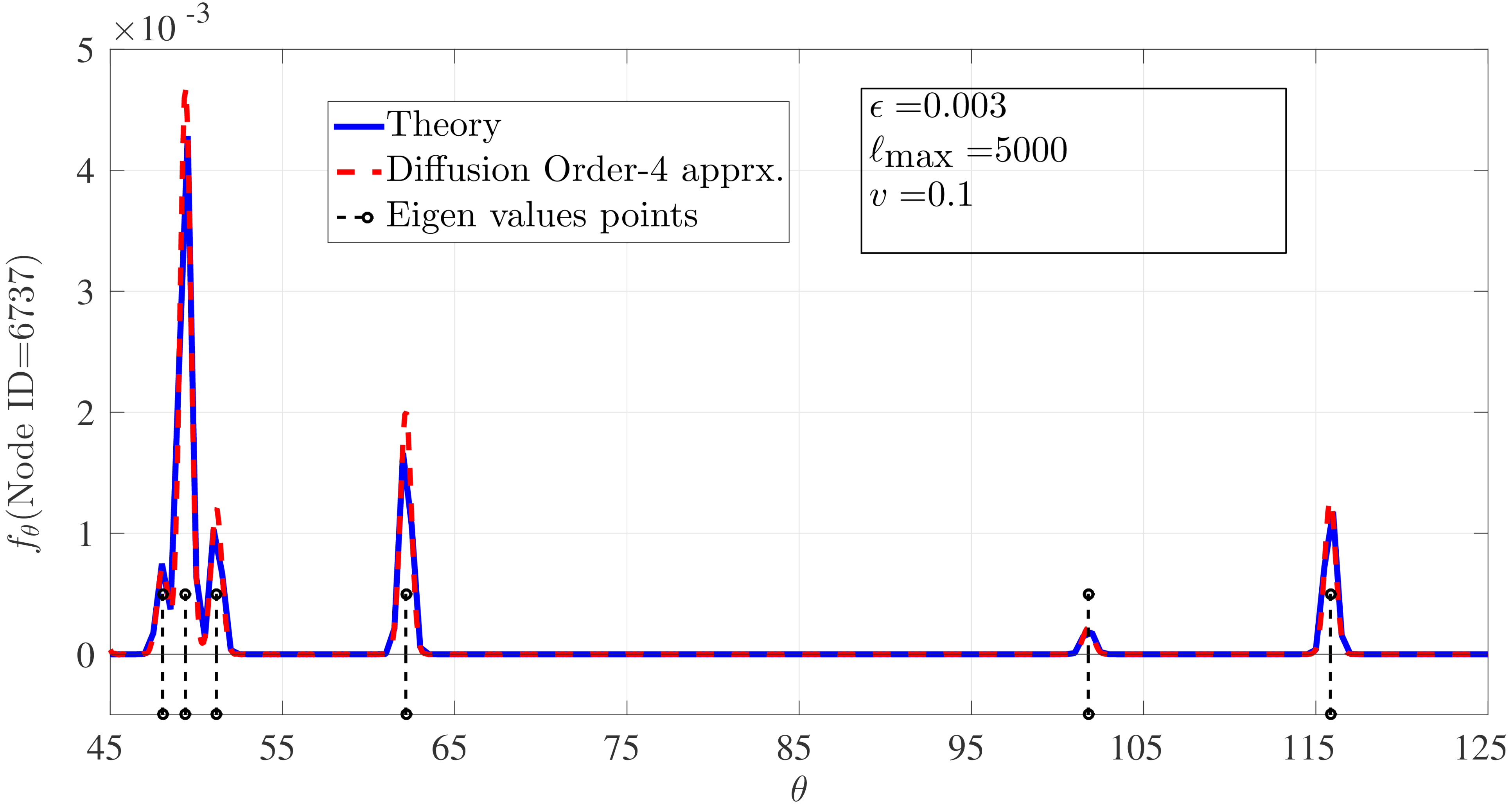


Monte Carlo gossiping

Parallel random walk

DBLP network

Number of nodes: 317K, number of edges: 1M.



Conclusions: Distributed Spectral Detection

- A simple interpretation of spectrum in terms of peaks at eigenvalue points.
- Developed distributed algorithms at node level based on complex power iterations
  - Complex diffusion: each node collects fluid from all the neighbors
  - Complex gossiping: each node collects fluid from one random neighbor
  - Parallel random walk implementation
- Connection with quantum random walk techniques
- Derived order of convergence and algorithms are scalable with the maximum degree of the graph
- Extension of algorithms to tackle higher resolution
- Numerical simulations on various real-world networks

- Topic 1: Spectral Decomposition: Sampling in “Spectral Domain”
- Topic 2: Network Sampling with Random Walk techniques

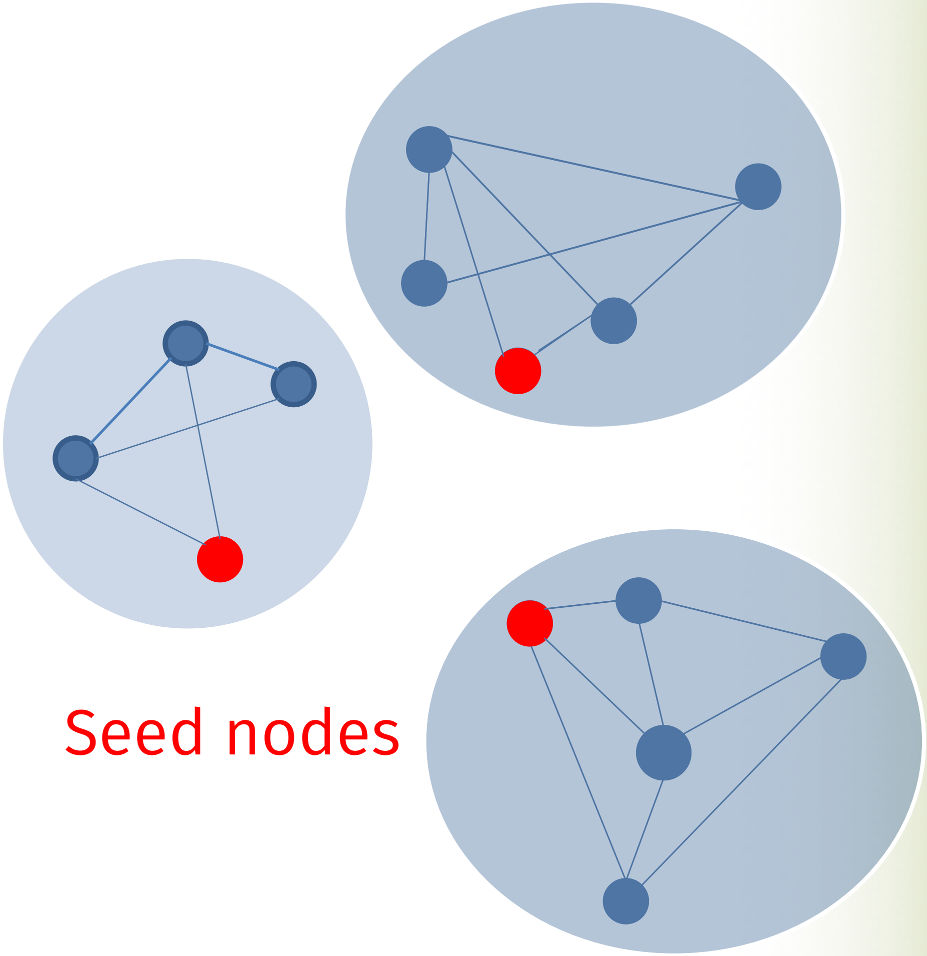
Motivation

- Online Social Network (OSN) users more likely to form edges with those with similar attributes?
- What proportion of a population supports a given political party?
- Average age of users in an OSN

Problem definition

Let $G = (V, E)$

- Undirected graph
- Node and edge have labels
- Not necessarily connected or has included connected components of interest
- Few seed nodes
- Large graph



Problem definition (contd.)

Estimate $\mu(G) = \sum_{(u,v)\in E} g(u,v)$

- Graph is unknown
- Only local information available

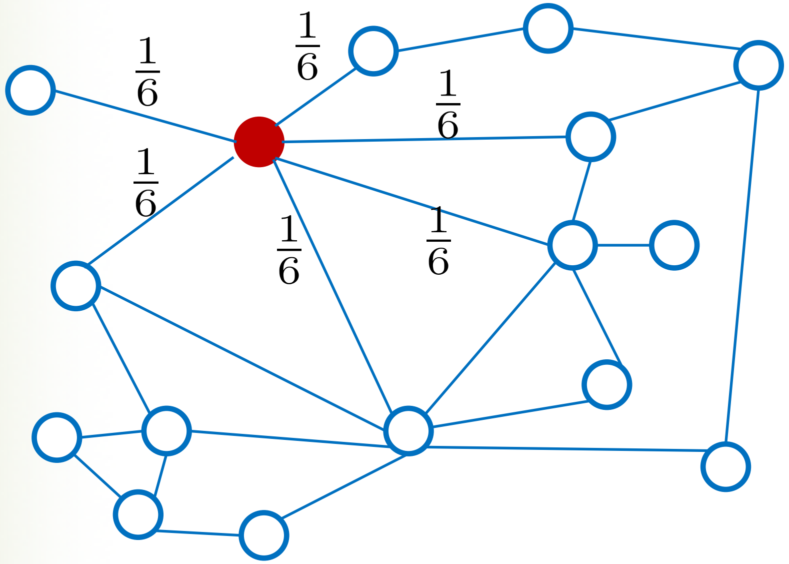
$\left\{ \begin{array}{l} \text{Seed nodes and their neighbor IDs} \\ \text{Query (visit) a neighbor} \\ \text{Visited nodes and their neighbor IDs} \end{array} \right.$

Solution: Sample all the neighbors (snow-ball sampling) ? ?

No, biased towards principal eigenvector. Exponential number of samples required

How do we know in real time if our estimates are accurate?

Random walk based estimation



Random walk $\{X_k\}_{k\geq 1}$ has unique stationary distribution $\{\pi_i\}_{i=1}^n$ if graph G is connected and non-bipartite

- Goal:

  Estimate $\mu(G) = \sum_{(u,v)\in E} g(u,v)$

- How [Ribeiro and Towsley `10]:

  Estimator for $\sum_{(u,v)\in E} g(u,v) : \frac{2|E|}{k} \sum_{i=1}^{k-1} g(X_i, X_{i+1})$

Asymptotically converges

Extensions: [Lee et al. `12], [Gjoka et al. `11] [Ribeiro et al. `12]

We get an estimate of $\mu(G)$ but how accurate is it ?

Network Sampling with Random Walk

Idea of tours

Properties of tours:

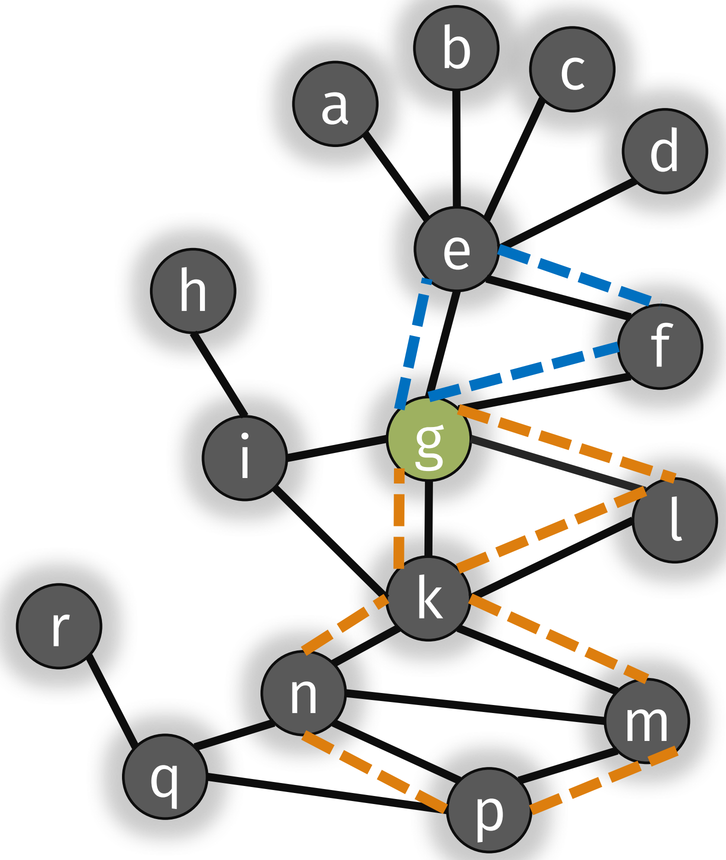
- Tours are independent
- Fully distributed crawler implementation

Issues with tours:

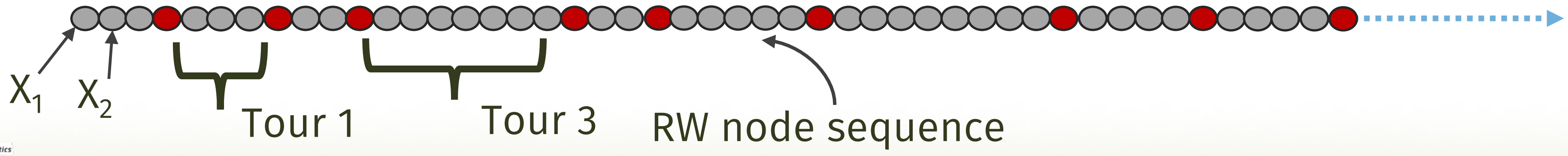
- Returning to same node will take “forever” in a large network [Massoulié et al’06]

$$\mathbb{E}[\text{Tour length}] = \frac{\overset{2|E|}{\mathrm{vol}(G)}}{\text{degree(g)}}$$

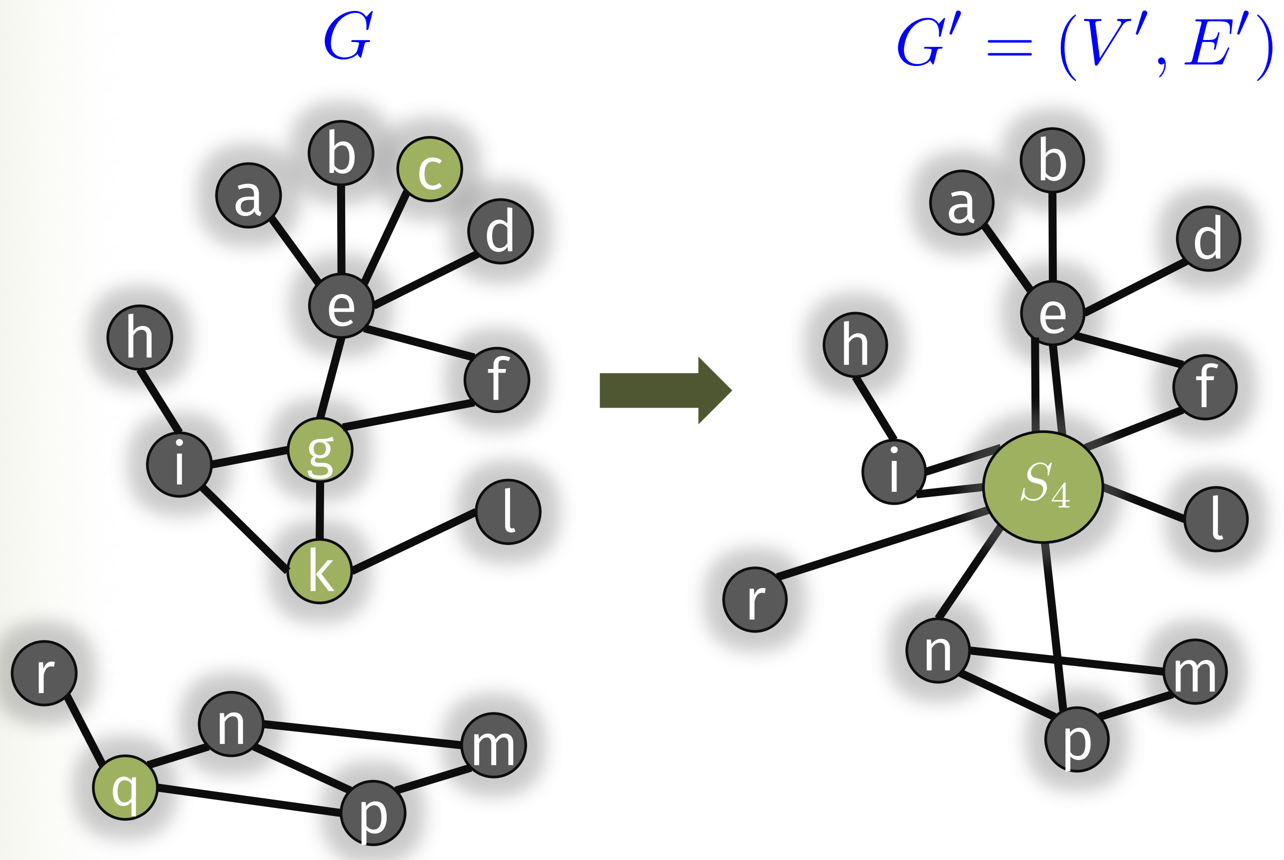
- Solution? Renewal from the most frequent node.
  - No, tours will be interdependent



● : most frequent node in sequence



The idea of Super-node



- Tackling disconnected graph
- Faster estimate with shorter crawls

$$\mathbb{E}[\text{Tour length}] = \frac{\text{vol}(G)}{\text{degree}(S_4)}$$

Super-node formation:

- static and dynamic (will see later)

Estimator

Length of kth tour

True value of the contracted graph

Key property of tours: $$\mathbb{E}\left[\sum_{t=2}^{\xi_k} f(X_{t-1}^{(k)}, X_t^{(k)})\right] = \frac{2}{d_{S_n}}\mu(G')$$

$f(u,v) := g(u,v)$
except when u or v is S_n

Samples in kth tour

Degree of super-node

$$\hat{\mu}(G) = \overbrace{\frac{d_{S_n}}{2m}\sum_{k=1}^{m}\sum_{t=2}^{\xi_k} f(X_{t-1}^{(k)}, X_t^{(k)})}^{\text{Estimate from crawls}} + \overbrace{\sum_{(u,v)\in H} g(u,v)}^{\substack{\text{Given knowledge} \\ \text{from nodes in super-node}}}$$

Induced subgraph from the nodes inside S_n

Estimator

- Unbiased (unlike asymptotic in [Ribeiro and Towsley '10])

$$\mathbb{E}[\hat{\mu}(G)] = \mu(G)$$

- Strongly consistent

$$\hat{\mu}(G) \to \mu(G) \text{ a.s.}$$

Confidence interval

$$P\left(|\mu(G) - \hat{\mu}(G)| \leq \varepsilon\right) \approx 1 - 2\Phi\left(\frac{\varepsilon\sqrt{m}}{\hat{\sigma}_m}\right)$$

Sampled variance

$$\mathrm{Var}\left[\sum_{t=2}^{\xi_k} f(X_{t-1}^{(k)}, X_t^{(k)})\right] \leq B^2\left(\frac{2\mathrm{vol}(G)}{d_{S_n}^2\,\delta'} + 1\right)$$

$\delta' :=$ spectral gap of new graph

$$\max_{(i,j)\in E'} f(i,j) \leq B < \infty$$

Bayesian formulation

Find a posterior probability distribution

$$\mathbb{P}(\mu(G) < x | \{m \text{ tours}\})$$

with a suitable prior distribution

Bayesian formulation (contd.)

Setting:

- Available no. of tours $= m$
- Divide m tours into $\sqrt{m}$ batches
- Let $\hat{F}_h$ be the estimate of $\mu(G)$ in h-th batch.
- Assumption: $\hat{F}_h \sim \mathrm{Normal}(\mu(G), \sigma^2)$

  (also justifiable via exponentially bounded tour lengths [Aldous and Fill '02])
- Assume priors $\mu(G)|\sigma^2 \sim \mathrm{Normal}(\mu_0, \sigma^2/m_0)$, $\sigma^2 \sim \text{Inverse-gamma}(\nu_0/2, \nu_0\sigma_0^2/2)$

Then for large values of m ($m \geq 2$),

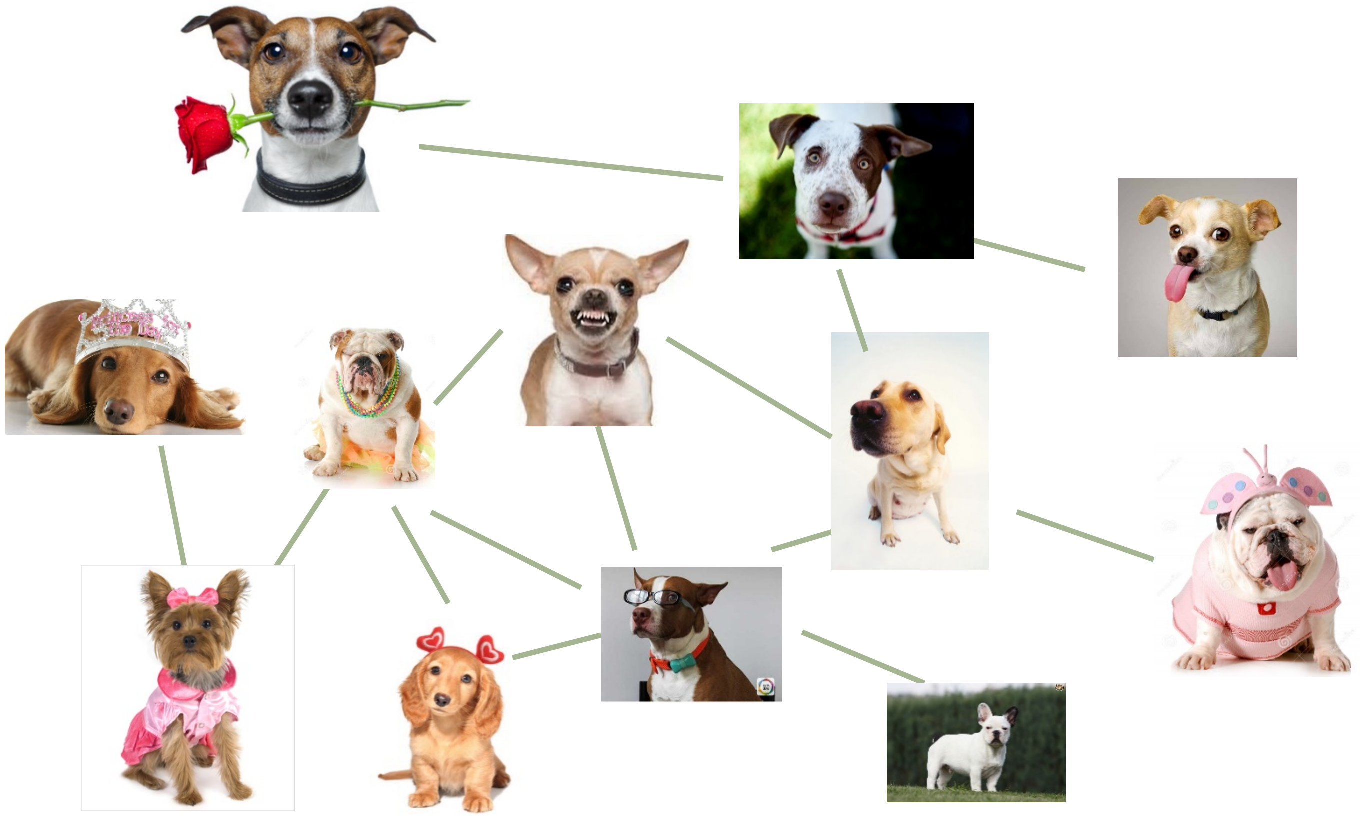
$$\mathbb{P}(\mu(G) \leq x | \{m \text{ tours}\}) \approx \phi_{\underset{(\nu,\widetilde{\mu},\widetilde{\sigma})}{\text{student-t}}}(x)$$

$$\nu = \nu_0 + \lfloor\sqrt{m}\rfloor,$$

$$\widetilde{\mu} = \frac{m_0\mu_0 + \lfloor\sqrt{m}\rfloor\hat{\mu}(G))}{m_0 + \lfloor\sqrt{m}\rfloor}, \widetilde{\sigma}^2 = \frac{\nu_0\sigma_0^2 + \sum_{k=1}^{\lfloor\sqrt{m}\rfloor}(\hat{F}_k - \hat{\mu}(G))^2 + \frac{m_0\lfloor\sqrt{m}\rfloor(\hat{\mu}(G))-\mu_0)^2}{m_0+\lfloor\sqrt{m}\rfloor}}{(\nu_0 + \lfloor\sqrt{m}\rfloor)(m_0 + \lfloor\sqrt{m}\rfloor)}$$

Simulations on real-world networks

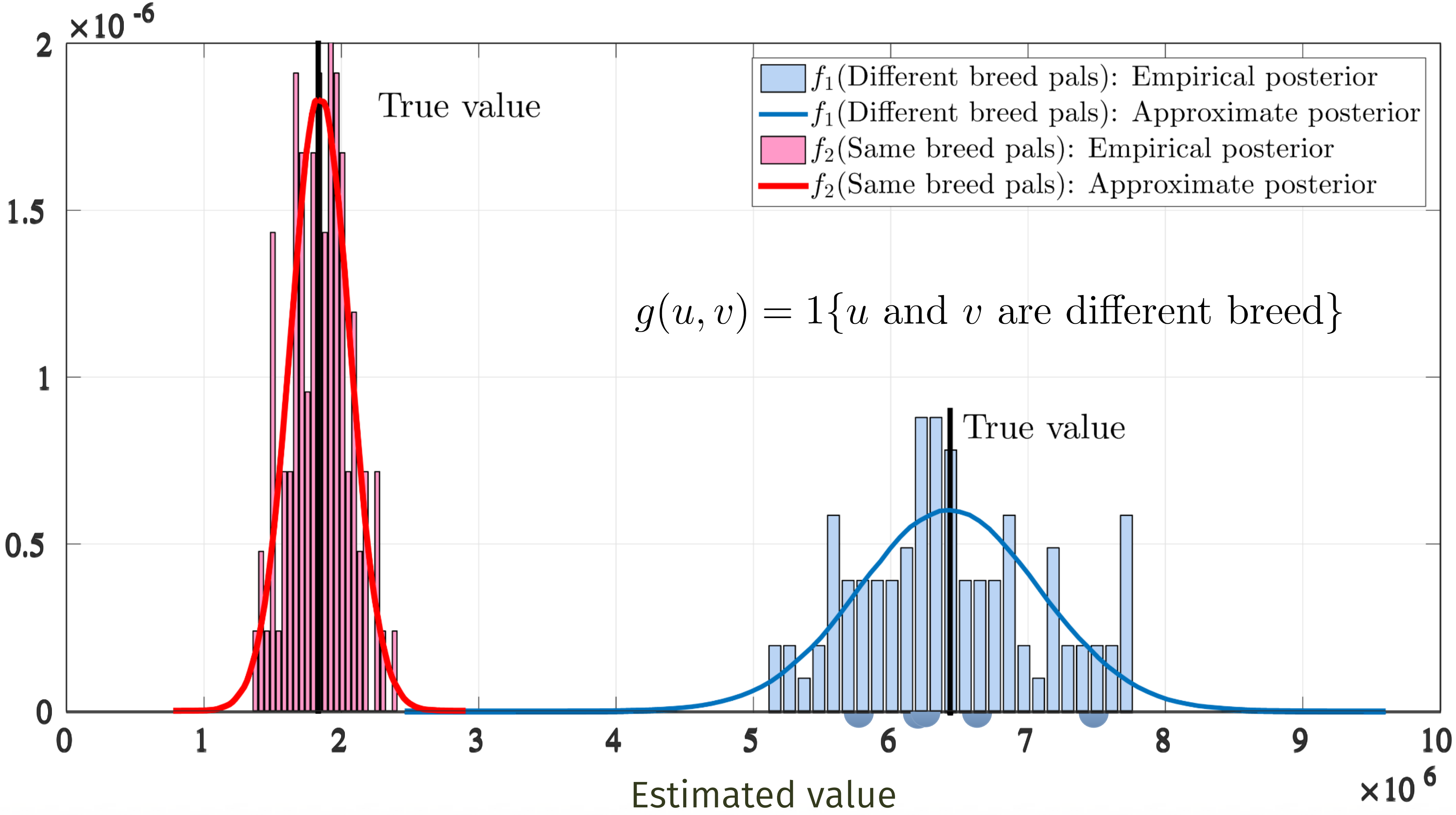
Dogster network: Online social network for dogs ?



Dogster network

415K nodes, 8.27M edges

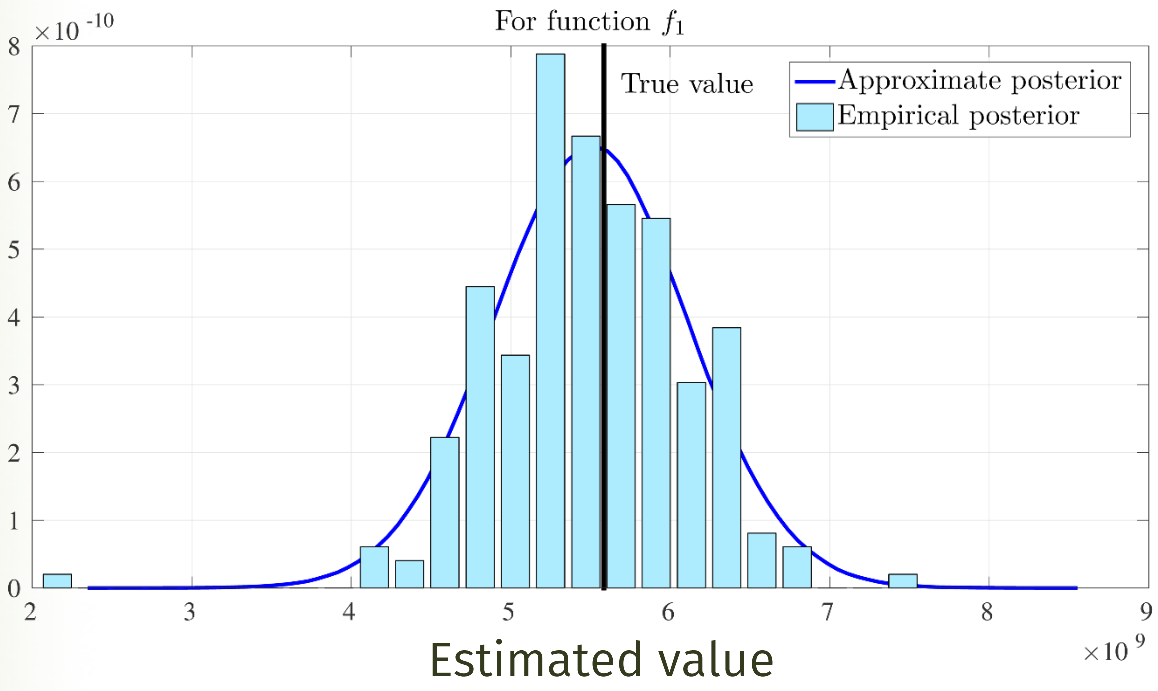
Percentage of graph covered: 2.72% (edges), 14.86% (nodes)

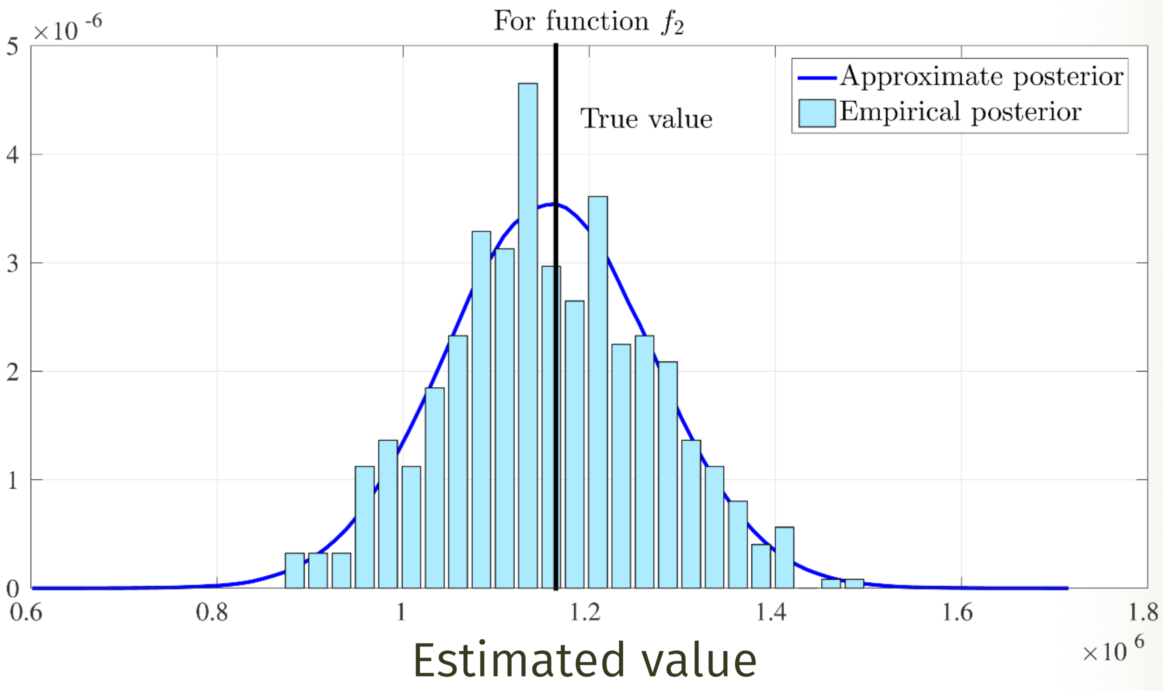


Friendster network

64K nodes, 1.25M edges

Percentage of graph covered: 7.43% (edges), 18.52% (nodes)





$$f_1 = d_{X_t}.d_{X_{t+1}}$$

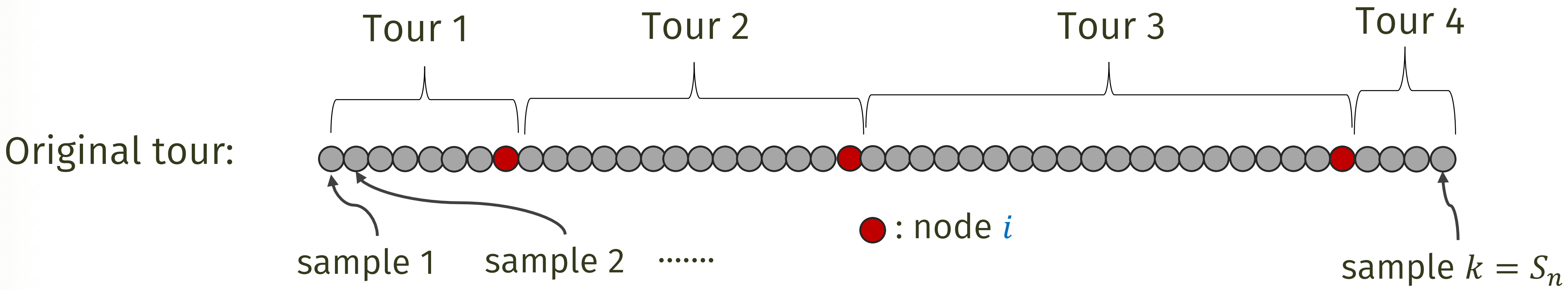
$$f_2 = \begin{cases} 1 \text{ if } d_{X_t} + d_{X_{t+1}} > 50 \\ 0 \text{ otherwise} \end{cases}$$

What if the super-node is not that “super”?

Adaptive crawler: super-node gets bigger as crawling progresses

How to add nodes to super-node:

1. via **any** method as long as independent of already observed tours
2. Emulate presence of new node i in super-node S_n from the start
   - Check previous tours. Break them when i is found.

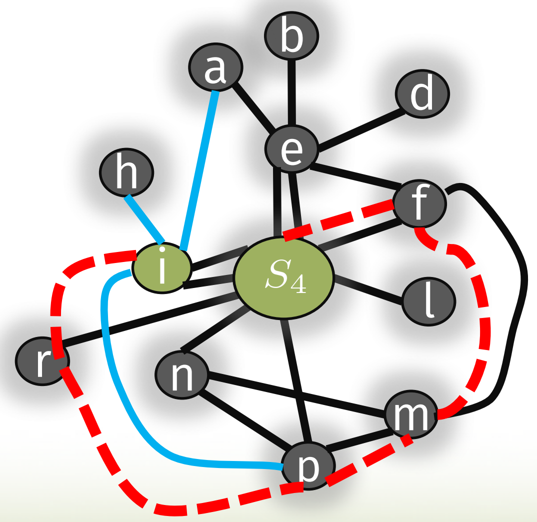


What if the super-node is not that “super”?

Adaptive crawler: super-node gets bigger as crawling progresses

How to add nodes to super-node:

1. via **any** method as long as independent of already observed tours
2. Emulate presence of new node i in super-node S_n from the start
   - Check previous tours. Break them when i is found.
   - Start k new tours from newly added node i;
     k ~ negative Binomial (function of degrees of i, S_n & no of tours)



What if the super-node is not that “super”?

Adaptive crawler: super-node gets bigger as crawling progresses

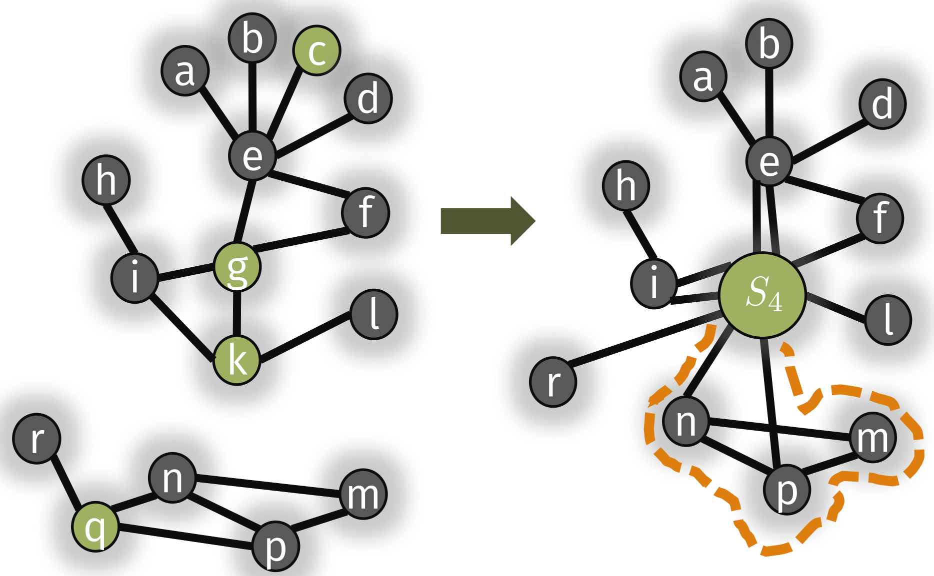
How to add nodes to super-node:

1. via **any** method as long as independent of already observed tours
2. Emulate presence of new node i in super-node S_n from the start
   - Check previous tours. Break them when i is found.
   - Start k new tours from newly added node i;
     k ~ negative Binomial (function of degrees of i, S_n & no of tours)

Theorem
Dynamic and static super-node sample paths are equivalent in distribution

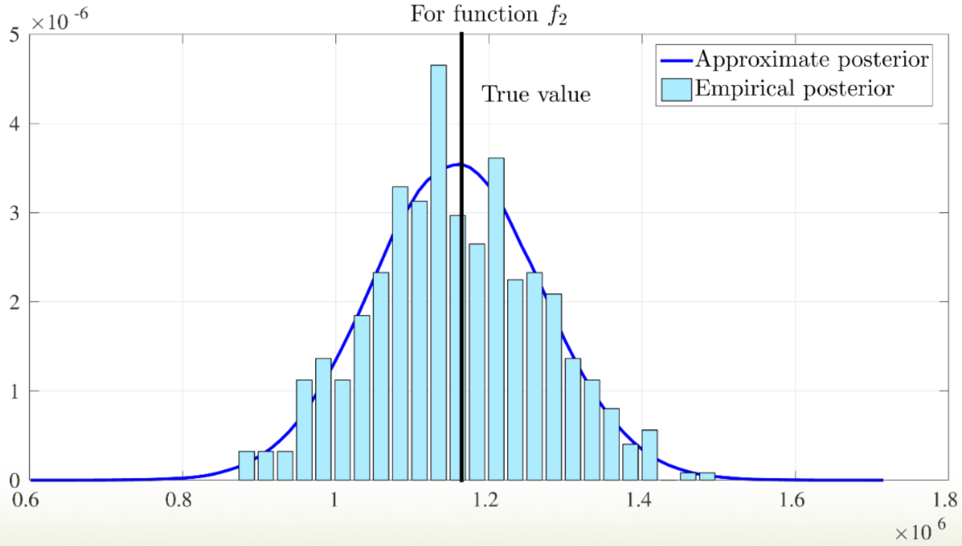
Conclusions: Network Sampling with Random Walk

- Unbiased estimator of $\mu(G) = \sum_{(u,v)\in E} g(u,v)$
- Propose dynamic super-node:
  - ✓ Short parallel random walk crawls
  - ✓ Parameter-free crawling



Conclusions: Network Sampling with Random Walk

- Unbiased estimator of $\mu(G) = \sum_{(u,v)\in E} g(u,v)$
- Propose dynamic super-node:
  - ✓ Short parallel random walk crawls
  - ✓ Parameter-free crawling
- Provides real-time assessment of estimation accuracy:
  - ✓ Bayesian formulation: posterior distribution, matches well true histogram



Conclusions: Network Sampling with Random Walk

- Unbiased estimator of $\mu(G) = \sum_{(u,v) \in E} g(u,v)$
- Propose dynamic super-node:
  - ✓ Short parallel random walk crawls
  - ✓ Parameter-free crawling
- Provides real-time assessment of estimation accuracy:
  - ✓ Bayesian formulation: posterior distribution, matches well true histogram
- If the given network forms connections randomly with same node attributes and degrees:
  - ✓ Estimation of expected value and variance of $\mu(G_{\mathrm{conf}})$

    $\mu(G)$ of the generated random graph model
  - ✓ Check whether original network value samples from distribution of $\mu(G_{\mathrm{conf}})$
- Reinforcement-Learning based Sampling: more stable and no burn-in!

Some Research Directions

Extension of spectral decomposition algorithms:

- Non-symmetric matrices
- Parameter-free or automatic procedure for the identification of eigenelements
- Why does Monter Carlo gossiping converge quickly?

Random walk based sampling:

- Asymptotic variance looks not much studied in literature, compared to mixing time, especially in case of random walk sampling in random graphs.
- Studies on the formation of super-node, effect of super-node selection in the asymptotic variance
- Concentration result for the tour based estimator providing time and memory complexities
- Reinforcement learning needs further study.

Extreme value theory:

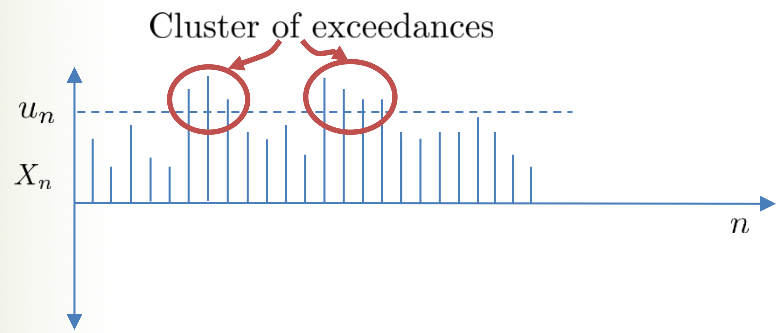
- Relation between extremal index and clustering and assortativity coefficient
- Deriving extremal index for more general graph correlation models.

Thank You!

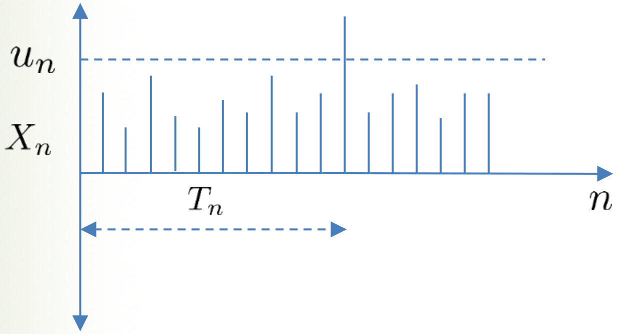
Extra slides

- Topic 1: Spectral Decomposition: Sampling in “Spectral Domain”
- Topic 2: Network Sampling with Random Walk techniques
- Topic 2: Extreme Value Theory and Network Sampling Processes

Questions We Address Here…



Statistical properties of clusters



First passage time

Kth largest value of samples and many more extremal properties

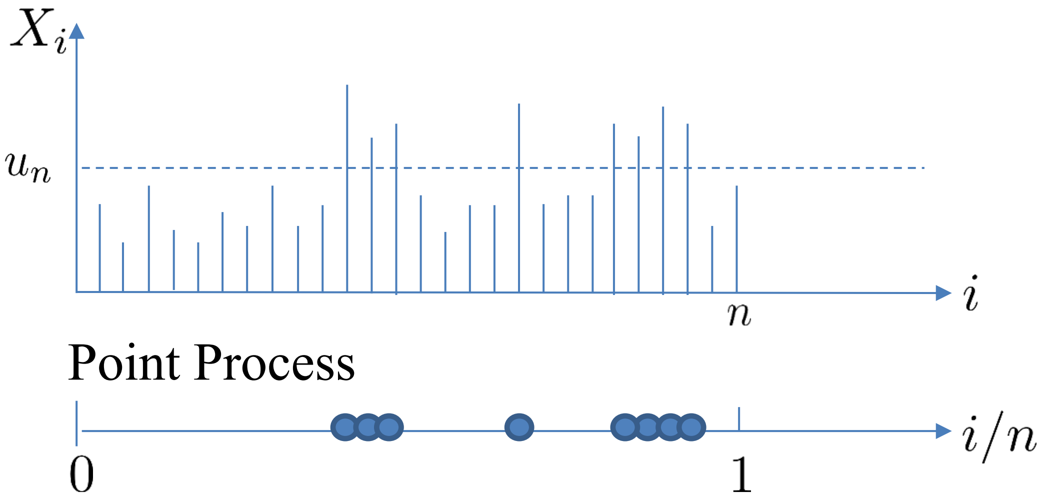
Is there a simple way to get information about many extremal properties? Ans: Extremal Index

Relation to Extreme Value Theory

Extremal Index (θ):

Defintion: If $\lim_{n\to\infty} E[\text{no of exceedances}] = \tau$,

$$P[\max(X_1, \ldots, X_n) \leq u_n] \to \exp(-\tau\theta)$$



Point process of exceedances →Compound poisson process (rate $\theta\tau$)

$$N_n(.) = \sum_{i \in \mathcal{I}} \mathbf{I}(i/n \in .), \mathcal{I} = \{i : X_i > u_n, 1 \leq i \leq n\} \qquad N_n \xrightarrow{d} CP(\theta\tau, \pi)$$

Extremal Index: Applications

- Gives maxima of the degree sequence with certain probability

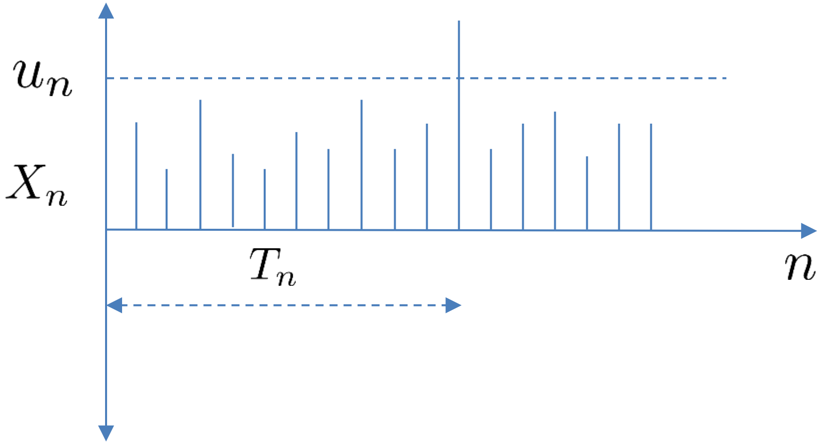
$$P\{\max\{X_1,\ldots,X_n\} \le x\} = F^{n\theta}(x) + o(1), n \to \infty$$

Pareto case revisited:

- i.i.d. degrees, largest degree$\approx KN^{1/\gamma}$, N no. of nodes, γ tail index of Pareto distribution (N. Litvak, LNCS'12)
- Stationary degree samples with EI, largest degree$\approx K(N\theta)^{1/\gamma}$

Extremal Index: Applications

First passage time:



$$\frac{T_n}{n} \text{ asymptotically} \sim \text{Exp}(\theta\tau)$$
$$E(T_n) \approx n/(\theta\tau)$$

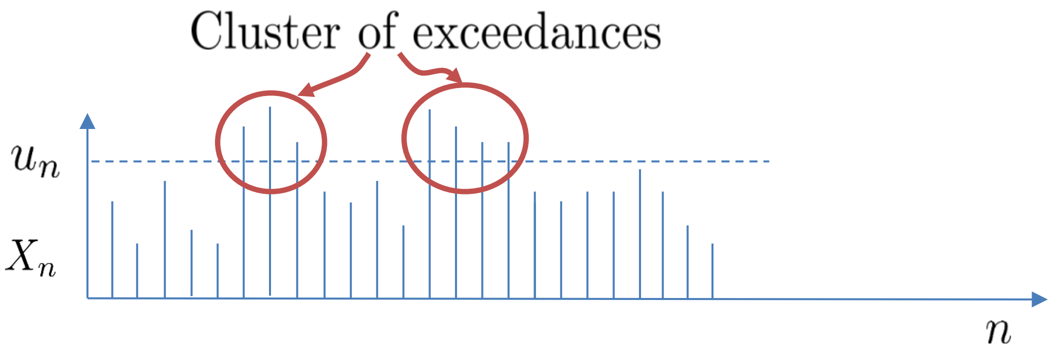
Lower the value of EI, more time to hit extreme levels

e.g. Pareto $P(X_i > u_n) = u_n^{-\alpha}, u_n = (n)^{1/\alpha}$ for $\tau = 1$

$$\implies E[T_n] \approx \frac{u_n^{\alpha}}{\theta}$$

Extremal Index: Applications

Relation to Mean Cluster Size:



$$\lim_{n\to\infty} E[\text{cluster size with } n \text{ samples}] = \frac{1}{\text{Extremal Index}}$$

Calculation of Extremal Index

Two mixing conditions on the samples

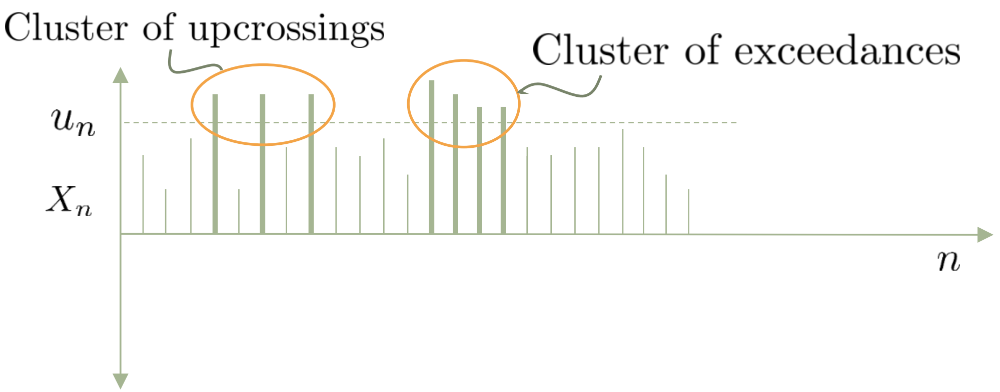
Cond-1: Limits long range dependence

$$|P(\mathcal{A}\mathcal{B}) - P(\mathcal{A})P(\mathcal{B})| \leq \alpha_n \quad \mathcal{A} \text{ and } \mathcal{B}\text{: events} \subset \{X_i \leq u_n\},\ l_n \text{ seperated}$$

$$l_n = o(n), \alpha_n \to 0$$

Stationary Markov samples or its measurable functions satisfy this

Cond-2:



$$\lim_{n\to\infty} P(\text{Cluster of upcrossings}) < \epsilon$$

$$\lim_{n\to\infty} P(\text{Cluster of exceedances}) \geq (1-\epsilon)$$

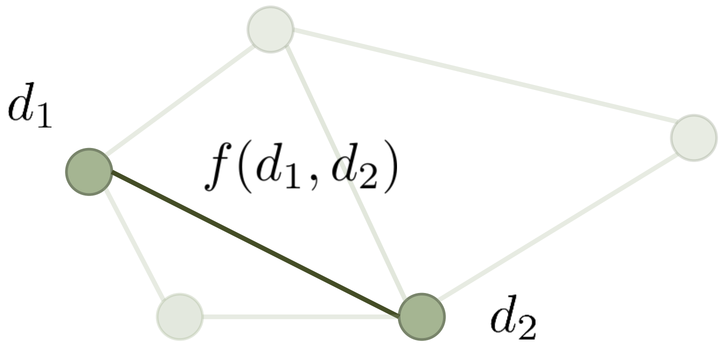
Proposition

If the sampled sequence is stationary and satisfies mixing conditions, then Extremal Index

$$\theta =< \mathbf{1}, \nabla C > \big|_{u=v=1} - 1,$$

$0 \leq \theta \leq 1$ and $C(u,v) = P(X_1 \leq F^{-1}(u), X_2 \leq F^{-1}(v))$ is the Copula.

Degree Correlations



- Undirected and correlated
- $f(d_1, d_2)$ is enough to construct graph

- Crawling via Random Walks on vertices
- Degree sequence is a Hidden Markov chain
- What is the joint stationary distribution on degree state space?

Generation of a Correlated Graph

Tail distribution $\bar{F}(d_1, d_2)$ is given.

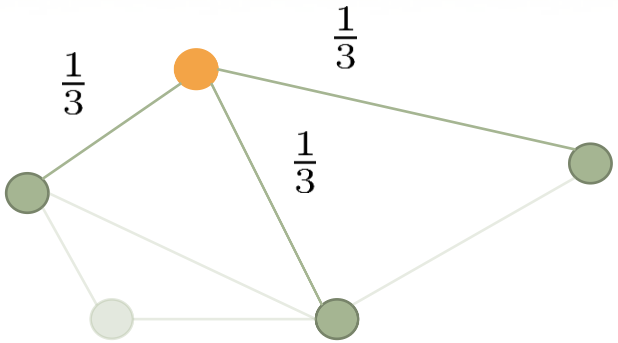
1. Degree sequence:

$$f_d(d) = \frac{f(d)E[D]}{d}$$

2. Uncorrelated random graph generation with configuration model
3. MCMC Metropolis-Hastings dynamics:
   a) Select 2 edges randomly: (v_1, w_1) and (v_2, w_2), degrees (j_1, k_1) and (j_2, k_2)

   b) With prob $\min\left(1, \frac{f(j_1, j_2) f(k_1, k_2)}{f(j_1, k_1) f(j_2, k_2)}\right)$, rewire edges to (v_1, v_2) and (w_1, w_2)

Meanfield Models



Standard Random Walk

$$f_{RW}(d_{t+1}|d_t) \approx \frac{1}{d_t} \cdot \frac{E[D] f(d_t, d_{t+1})}{f_d(d_t)}$$

$$f_{RW}(d_{t+1}, d_t) \approx f(d_{t+1}, d_t)$$

$P(\text{head}) = c$

Page Rank

with c, follow RW
with $1 - c$, uniform node sampling

$$f_{PR}(d_{t+1}|d_t) \approx c f_{RW}(d_{t+1}|d_t) + (1-c) f_d(d_{t+1})$$

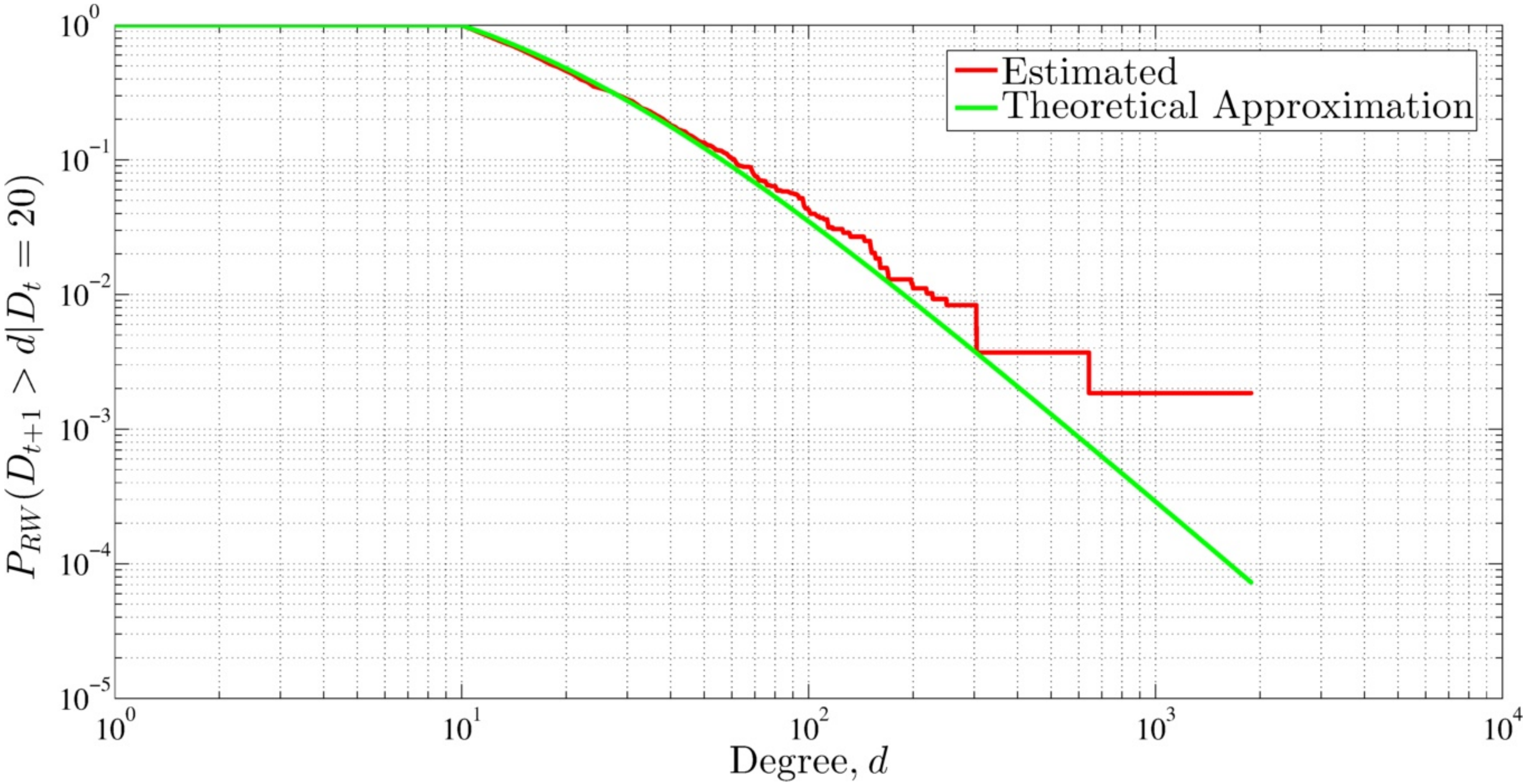
$P(\text{head}) = c$

Random Walk with Jumps (RWJ)

$$c = \frac{d_t}{d_t + \alpha}$$

$$f_{RWJ}(d_{t+1}, d_t) \approx \frac{E[D]}{E[D] + \alpha} f(d_{t+1}, d_t) + \frac{\alpha}{E[D] + \alpha} f_d(d_{t+1}) f_d(d_t)$$

Check of Meanfield Model in Random Walks



Degree correlation among neighbours, bivariate Pareto distributed

$$\bar{F}(d_1, d_2) = \left(1 + \frac{d_1 - \mu}{\sigma} + \frac{d_2 - \mu}{\sigma}\right)^{-\gamma} \quad \mu = 10, \sigma = 15, \gamma = 1.2$$

Extremal Index for Bivariate Pareto Model

$$\bar{F}(d_1, d_2) \sim \left(1 + \frac{d_1 - \mu}{\sigma} + \frac{d_2 - \mu}{\sigma}\right)^{-\gamma}$$

$$\text{Random Walk: } \mathrm{EI} = 1 - 1/2^{\gamma}$$

$$\text{Random Walk with Jumps: } \mathrm{EI} = 1 - \frac{E[D]}{E[D] + \alpha} 2^{\gamma}$$

$$\text{PageRank: } \mathrm{EI} \geq (1 - c)$$

(for any kind of degree correlations)

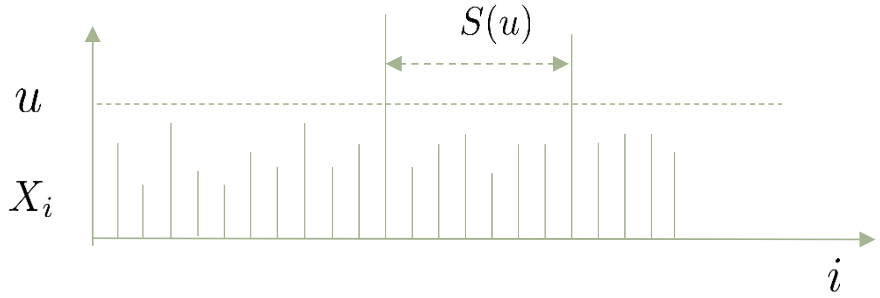
Estimation of Extremal Index

Empirical Copula based estimator:

$$C_n(u,v) = \frac{1}{n}\sum_{i=1}^{n}\mathbb{I}\left(\frac{R_i^X}{n+1} \le u, \frac{R_i^Y}{n+1} \le v\right)$$

EI: slope at (1; 1),Linear least square fitting & numerical differentiation

Intervals Estimator:



Based on

$$\bar{F}(u)S(u) \xrightarrow{d} T_\theta \quad T_\theta \sim \begin{cases} 1-\theta & \text{when } \theta = 0 \\ \mathrm{Exp}(\theta) & \text{when } \theta > 0 \end{cases}$$

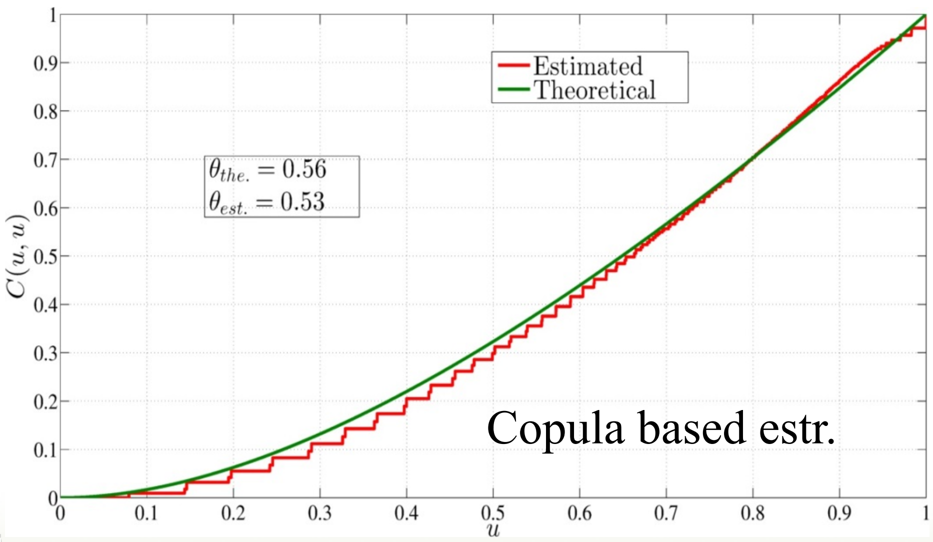
Use $E(T_\theta^2) = 2/\theta$ to obtain estimates

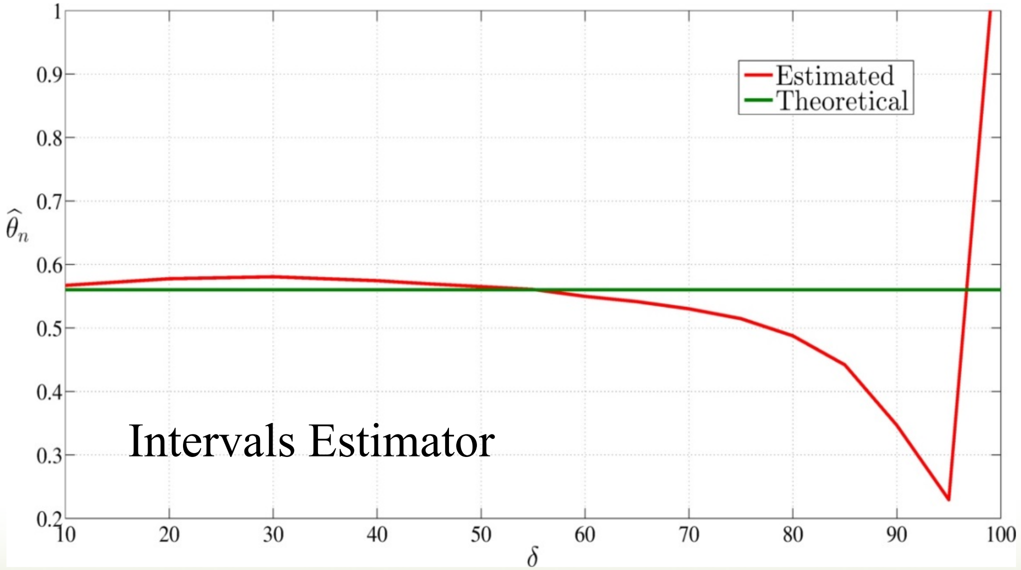
Numerical Results: Synthetic Graphs

Degree correlation between neighbours

$$\bar{F}(d_1, d_2) = \left(1 + \frac{d_1 - \mu}{\sigma} + \frac{d_2 - \mu}{\sigma}\right)^{-\gamma} \quad \mu = 10, \sigma = 15, \gamma = 1.2$$

EI	Analysis	Copula based estimator	Intervals Estimator
Synthetic graph (5K Nodes)	0.56	0.53	0.58





Numerical Results: Real Graphs

EI	Copula based estimator	Intervals Estimator
DBLP (32K Nodes,1.1M Edges)	0.29	0.25
Enron Email (37K Nodes,368K Edges)	0.61	0.62

Conclusions: Extreme Value Theory (Not presented)

- Associated Extremal Value Theory of stationary sequence to sampling of large graphs
- For any general stationary samples meeting two mixing conditions, knowledge of bivariate distribution or bivariate copula is sufficient to derive many extremal properties
- Extremal Index (EI) encapsulates this relation
- Applications of EI to many relevant extrems:
  - First hitting time, Order statistics, Mean cluster size
- Modeled correlation in degrees of adjacent nodes and random walk in degree state space
- Estimates EI for synthetic graph with degree correlations and find a good match with theory
- Estimated EI for two real world networks

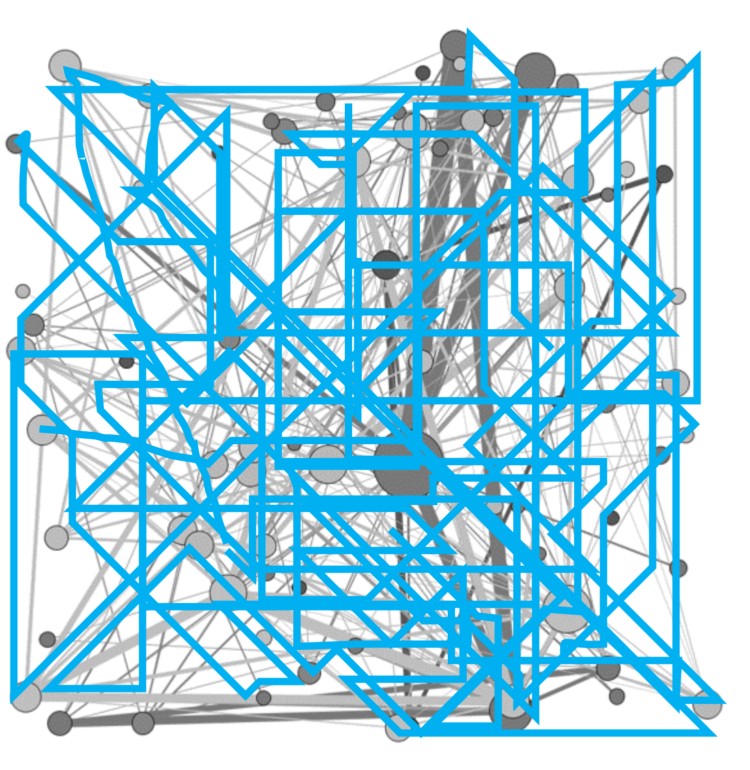
- Topic 1: Spectral Decomposition: Sampling in “Spectral Domain”
- Topic 2: Network Sampling with Random Walk techniques

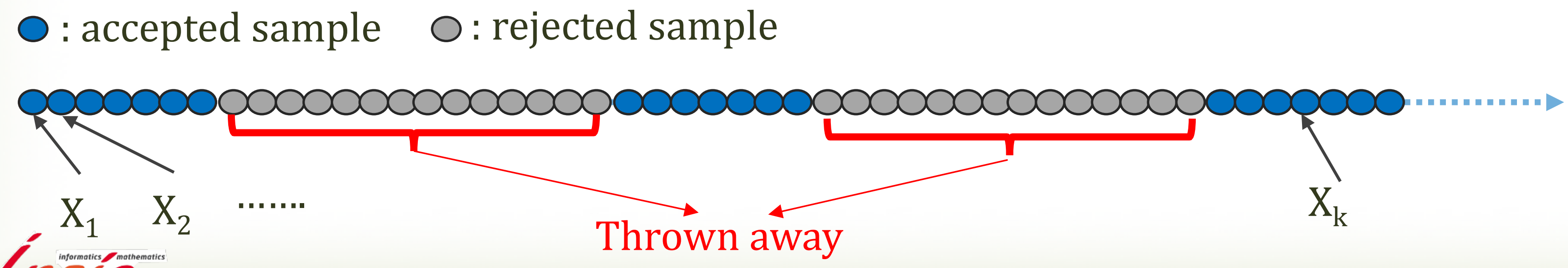
Existing asymptotic techniques and issues

- Asymptotic convergence: Ergodic theorem
  - Crawling the graph multiple times
- Variety of convergence diagnostics for MCMCs

Roughly divided into:

- Multiple walks to check convergence
  - Walks not independent (start at same seeds)
  - No guarantees
- Break a long walk into “nearly” independent segments
  - Asymptotic & throws away most observations





From metric $\mu(G)$ does network look random ?

Estimation and hypothesis testing in Chung-Lu or configuration model

Assumption: edges labels can be written as a function of node labels

- Does the true value of the given graph $\mu(G) = \sum_{(u,v)\in E} g(u,v)$ belongs to the class of values when the edges are formed purely at random?

$$\mu(G) \sim \text{ Distribution}(\mathbb{E}[\mu(G_{\text{random}})], \text{ Var}[\mu(G_{\text{random}})])$$

- Does the true value belongs to the class when the connections are formed based on degrees alone with no other influence ?

  Configuration model:

  - Assume the degree sequence same as that of G.
  - Edges formed by uniformly selecting the half edges of each node

Estimation in Chung-Lu or configuration model

Estimate $\mathbb{E}[\mu(G_{\text{conf}})]$ & $\text{Var}[\mu(G_{\text{conf}})]$

- The entire degree sequence unknown; only the degrees of sampled nodes known

$$\mathbb{E}[\mu(G_{\text{conf}})] = \sum_{\substack{(u,v)\in E\cup E^c \\ u\neq v}} g(u,v)\,\frac{d_u d_v}{2M} + \sum_{\substack{(u,v)\in E\cup E^c \\ u=v}} g(u,v)\,\frac{\binom{d_u}{2}}{2M}\,.$$

Random walk with jumps to estimate $g(u,v)$, for $(u,v) \notin E$

$$\Pr(\text{head}) := p = \frac{d_t}{d_t + \alpha}$$

with p, follow RW
with $1-p$, uniform node sampling

Hypothesis testing with the Chung-Lu model

$$\sum_{(u,v)\in E_{\text{C-L}}} g(u,v) \sim \text{Normal}\left(\mathbb{E}[\mu(G_{\text{C-L}})], \text{Var}(G_{\text{C-L}})\right)$$ (Lindeberg central limit theorem)

Look for the value of a the following satisfies

$$|\hat{\mu}(G) - \mathbb{E}[\mu(G_{\text{C-L}})]| \leq a\sqrt{\text{Var}(G_{\text{C-L}})}$$

Estimate value of given graph

Mean and variance of Chung-Lu graph

Dogster network: Estimator for $\mathbb{E}[\mu(G_{\text{C-L}})]$

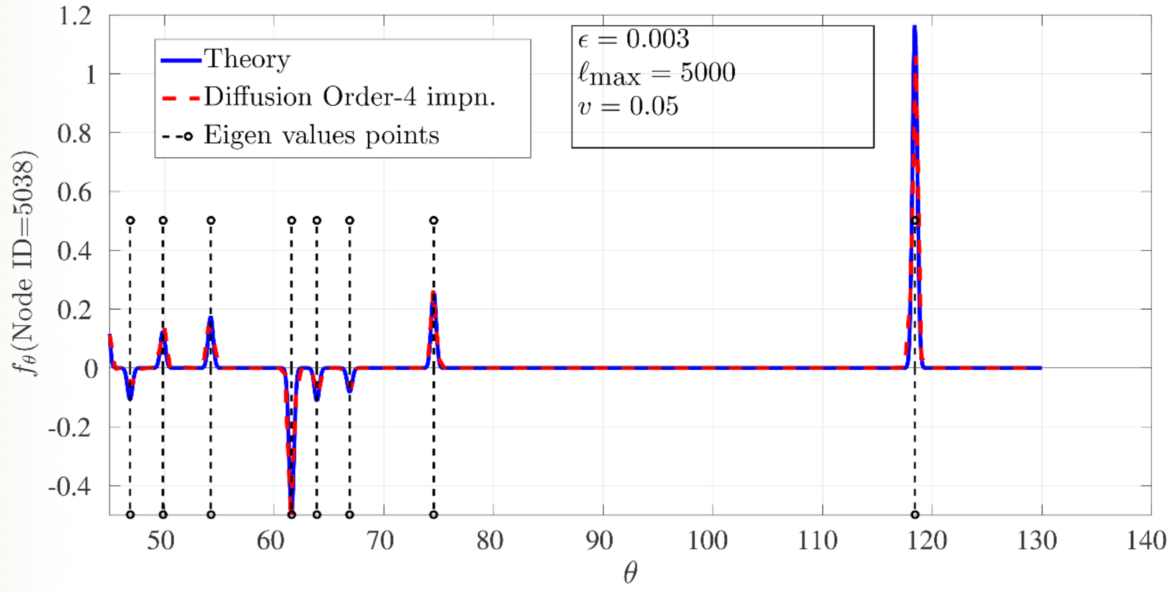
Percentage of graph crawled: 8.9% (edges), 18.51% (nodes)

Edge function	True value	Estimated value
1{same breed nodes}	8.12×10^{6}	8.066×10^{6}
1{different breed nodes}	2.17×10^{5}	1.995×10^{5}

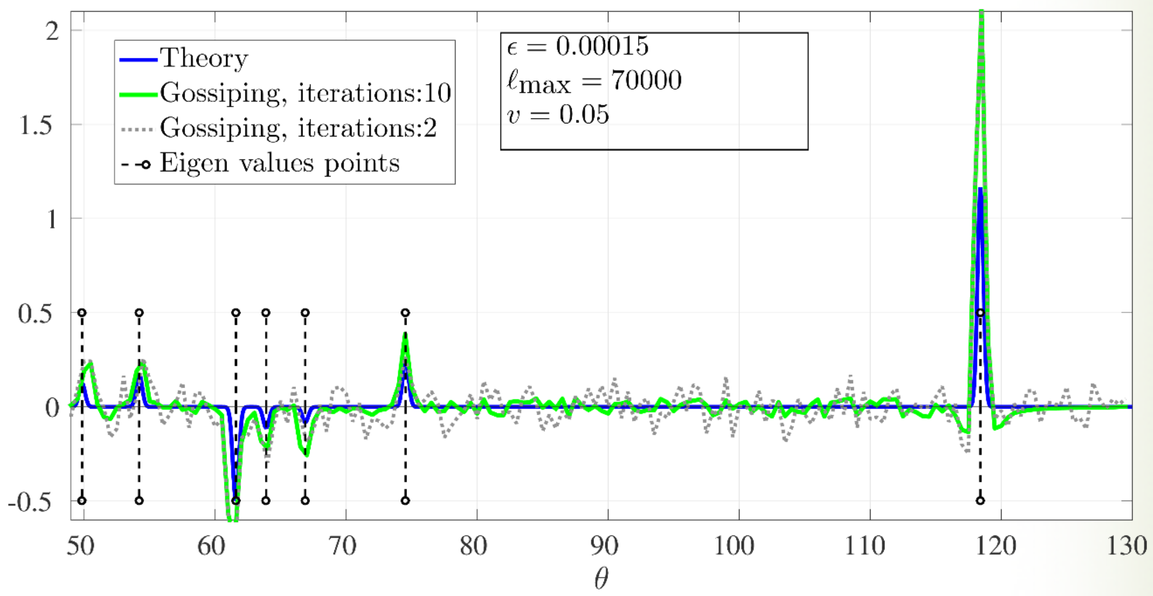
- Topic 1: Spectral Decomposition: Sampling in “Spectral Domain”
- Topic 2: Network Sampling with Random Walk techniques

Enron email network

Number of nodes: 33K, number of edges: 180K.



Complex diffusion order-4



Monte Carlo gossiping

Choice of parameters

Let Δ be the maximum degree.

1. Parameter v: With 99.7% of the Gaussian areas not overlapping,

$$6v < \min_{1 \leq i \leq k-1} |\lambda_i - \lambda_{i+1}| < 2\lambda_1 < 2\Delta$$

2. Parameter ε: From sampling theorem, to avoid aliasing,

$$\varepsilon < \frac{1}{2(|\lambda_1 - \lambda_n| + 6v)}$$

   Choosing $\varepsilon < \frac{1}{4\Delta + 12v}$ will ensure this.

3. Parameter $\ell_{\max}$: $1/\ell_{\max} < \varepsilon < 1/\sqrt{\ell_{\max}}$

Getting equal peaks for all eigenvalues

Take $\mathbf{b}_0$ as a vector of i.i.d. Gaussian$(0, w)$:

$$\mathbb{E}[\mathbf{b}_0^\intercal \mathbf{f}(\theta)] = w \sum_{j=1}^{n} \sqrt{\frac{2\pi}{v}} \exp(-\frac{(\lambda_j - \theta)^2}{2v})$$

- Detecting algebraic multiplicity

QRW Technique

1. Walker represented by a qubit with $\ell_{\max}$ atoms: Initialized as $(1/\sqrt{\ell_{\max}}) \sum_{k=0}^{\ell_{\max}-1} |k\rangle$.
2. State $|k\rangle$ gets delyed by $k\varepsilon$ time units via *splitting chain*
3. Walker moves as CT-QRW with wave function

$$\mathbf{\Psi}_t^{\ell_{\max}} = \frac{1}{\sqrt{\ell_{\max}}} \sum_{k=0}^{\ell_{\max}-1} e^{i(t-k\varepsilon)\mathcal{H}} \mathbf{\Psi}_0 |k\rangle.$$

4. At $t \geq \varepsilon \ell_{\max}$, on node m apply QFT on $\mathbf{\Psi}_t^{\ell_{\max}}(m) \implies \sum_{k=0}^{\ell_{\max}-1} y_k |k\rangle$
5. When we measure, we see k with probability $|y_k|^2$, an eigenvalue point shifted by Δ.

Different Approaches

1. **Centralized approach**: Adjacency matrix is fully known
2. **Our distributed approaches**
   - **Complex diffusion**: Asynchronous. Only local information available, communicates with all the neighbors
   - **Monte Carlo Gossiping**: Only local information, and communicates with only one neighbor.

   It can be implemented via parallel random walks as well